2016

UNIVERSIDAD COMPLUTENSE DE MADRID

JAVIER MATEO FERNÁNDEZ

[Análisis de contenidos en Social Media: Clasificación de mensajes e identificación de influyentes en el Banco Central Europeo (BCE).]

En este breve documento se esbozan los aspectos básicos del proyecto final de Máster en el área de la Minería de Datos e Inteligencia de Negocios.

Contenido

1. INTRODUCCIÓN 2
2. OBJETIVOS DEL PROYECTO 4
3.1 Objetivo principal 5
3.2 Objetivos secundarios 5
4. METODOLOGÍA, SOFTWARE Y CRONOGRAMA 6
4.1 Metodología de análisis 6
4.1.1 Extracción y Recolección de Datos 6
4.1.1.1 Acceso a los Datos de Twitter 7
4.1.1.2 Acceso a los Datos de Foros-Blogs 9
4.1.2 Recopilar y cotejar los conjuntos de Datos 9
4.1.3 Análisis de Sentimiento 11
4.1.4 Problemas de Centralidad y Liderazgo 12
4.1.5 Detección de Comunidades 13
4.1.6 Visualización y Creación de Dashboards 14
4.2 Programas de análisis 15
4.3 CRONOGRAMA 19
5. PROYECTO BANCO CENTRAL EUROPEO (BCE): 20
5.1 Extracción y Recolección de Datos: 20
5.1.1 Twitter, utilización de las API's de Twitter y extracción de datos con el paquete estadístico R 21
5.1.2 Foros y Blogs, utilización de la API creada en Python para la extracción de datos (Crawling&Scraping) 23
5.2 Recopilar y cotejar los conjuntos de Datos 30
5.3 Análisis de Sentimiento 32
5.4 Problemas de Centralidad y Liderazgo 41
5.4.1 Introducción 41
5.4.2 Teoría y Métricas 42
5.5 Detección de Comunidades 53
5.6 Visualización y Creación de Dashboards 58
6. CONCLUSIONES 60
7. VENTAJAS DEL USO DE SNA Y BIG DATA 61
8. BENEFICIOS DEL USO DE SNA 65
9.BIBLIOGRAFÍA 69

1. INTRODUCCIÓN

Con la llegada de las nuevas tecnologías y sus tendencias asociadas, la información que se genera en nuestra sociedad está creciendo de manera exponencial. Uno de los escenarios en los que la cantidad de información que se genera cada día empieza a ser inabordable es en el marco de las redes sociales. Esta nueva fuente de información ha atraído la atención de muchos profesionales que ven en las redes sociales una fuente inagotable de información para poder entender muchos de los problemas que nos preocupan hoy en día. Por este motivo, también ha crecido de manera significativa el análisis y el estudio de las redes sociales online. Ello es debido principalmente a la gran cantidad de usuarios que las utilizan, lo cual ha llevado a que exista una gran cantidad de información sobre productos, personas, gustos, eventos, marcas, opiniones políticas.

La gran cantidad de datos disponibles en Redes Sociales, Marketing, Finanzas, Economía ha permitido revelar la estructura interconectada de dichos datos entre sí. El análisis de redes sociales es el conjunto de herramientas analíticas y computacionales para estudiar estos datos interconectados. Cualquier persona sin o con conocimientos tecnológicos, se pregunta cómo se almacena toda la información que se genera en el mundo: en Facebook, Twitter, Smartcities o como Google es capaz de manejar todas las transacciones que se hacen a diario. El estudio de las redes sociales y el papel que juegan en este proceso de influencia social da lugar a varios tipos de aplicaciones como: Detectar personas influyentes y hacer marketing dirigido a ellas, para que influyan en las decisiones de las personas a las que los siguen (marketing viral); o detectar cambios en los entornos sociales de clientes y poder anticiparse a un cambio de comportamiento/consumo de los mismos.

Simplemente por mencionar una aplicación de este último proceso, tenemos el conocido ***Social CRM*** en el que las empresas utilizan las redes sociales online u otros datos de redes sociales para comprender mejor el comportamiento o influencia social de sus clientes. Uno de los sectores donde este tipo de análisis es más natural es en aquellas compañías que tienen datos sobre la interacción de sus propios clientes, como por ejemplo las empresas Telco (Telecomunicaciones), a partir de las llamadas entre clientes, Bancos, a partir de las transferencias entre clientes. Quizás el sector más avanzado en el uso del análisis de redes sociales en procesos de gestión de clientes es el sector de las Telco donde se utiliza para predecir la tasa de cancelación de clientes (*churn)*, detectar los clientes denominados "influencers" o "αα-users", mejorar el "targeting" en la adopción de productos, etc.

Además, el análisis de redes sociales sirve para identificar dos objetivos de gran interés; **Lo que hace la gente** (comportamiento de los usuarios) y **como se siente la gente** (sentimiento/opinión de los usuarios). Monitorizar ambos nos puede permitir predecir y gestionar diferentes procesos sociales, económicos o empresariales. Por ejemplo podemos,

- Conocer la opinión de millones de usuarios sobre una marca/persona/evento. De las propias y de los competidores.

    - Gestionar las quejas cuando sale un producto (customer service)
    - Medir el efecto de campañas en redes sociales (campaign measurement).
    - Gestionar la reputación y las crisis de reputación de la marca.

- Identificar a las personas relevantes en la conversación sobre un tema.

    - Hacer targeting sobre esos usuarios (influencer marketing)

- Identificar nuevos temas/gustos que nuestros clientes (market research).

    - Crear nuevos productos/servicios para atraer clientes.

Algunas aplicaciones directas derivadas de los problemas anteriores son: Las redes sociales como centro de atención a clientes[1;] usar las redes sociales para tener un mejor *credit scoring* de tus clientes. Compañías como Lenddo ya lo hacen[2]; usar las redes sociales para predecir/monitorizar las elecciones.[3]; mejorar la contratación de empleados.[4]; o predecir/monitorizar los mercados financieros[5].

Con el fin de poner al lector en situación, uno de los factores clave a la hora de analizar las Redes Sociales y blogs de Internet es la estructura de la información extraída. Dicha información es llamada formalmente como datos no estructurados.

A continuación se cita una posible definición sobre datos no estructurados: "son aquellos datos no almacenados en una base de datos tradicional. La información no estructurada no puede ser almacenada en estructuras de datos relacionales predefinidas."

Se pueden establecer diferentes clasificaciones, vamos a considerar dos de ellas.

- Datos no estructurados y semiestructurados. Los datos semiestructurados serían aquellos datos que no residen de bases de datos relacionales, pero presentan una organización interna que facilita su tratamiento, tales como documentos XML y datos almacenados en bases de datos NoSQL.
- Datos de tipo texto y no-texto. Datos no estructurados de tipo texto podrían ser datos generados en las redes sociales, foros, e-mails, presentaciones Power Point o documentos Word, mientras que datos no-texto podrían ser ficheros de imágenes jpeg, ficheros de audio mp3 o ficheros de video tipo flash.

[1] https://hootsuite.com/es/soluciones/social-customer-service
[2] http://www.forbes.com/sites/tomgroenfeldt/2015/01/29/lenddo-creates-credit-scores-using-social-media/
[3] http://portal.uc3m.es/portal/page/portal/actualidad_cientifica/noticias/conversacion_politica_en_twitter
[4] http://hiring.monster.com/hr/hr-best-practices/recruiting-hiring-advice/acquiring-job-candidates/social-media-recruiting-guidelines.aspx
[5] https://www.rt.com/usa/272758-twitter-predict-stock-market/

Las principales características de los datos no estructurados son las siguientes:

- Volumen y crecimiento: el volumen de datos y la tasa de crecimiento de los datos no estructurados es muy superior al de los datos estructurados. Por ejemplo, twitter genera 12 Terabytes de información cada día. De acuerdo con Gartner, la tasa anual de crecimiento de datos es del 40 a 60 por ciento, pero para los datos no estructurados en empresas, la tasa de crecimiento puede llegar al 80 por ciento (informe 2012).
- Orígenes de datos: El origen de los datos es muy diverso: datos generados en redes sociales, datos generados en foros, e-mails, datos extraídos de la web empleando técnicas de web semántica, documentos internos de la compañía (word, pdf, ppt).
- Almacenamiento: Debido a su estructura no podemos emplear arquitectura relacional, siendo necesario trabajar con herramientas 'Big Data', siendo crítico en estas arquitecturas los aspectos relacionados con la escalabilidad y paralelismo. Según el tipo de dato se impone el almacenamiento cloud. Monitorizar la frecuencia de uso y la detección de datos inactivos son aspectos críticos de cara a reducir costes de almacenamiento.
- Terminología e idiomas: La terminología es una cuestión crítica tratando datos no estructurados de tipo texto. Es habitual llamar a lo mismo de diferentes formas, de tal modo que es necesario una racionalización de la terminología. Otra cuestión es el idioma en el que se he generado la información tratada.
- Seguridad: Hay que considerar que algunos datos no estructurados de tipo texto, pueden no ser seguros. Por otra parte el control de accesos a los mismos es complejo debido a cuestiones de confidencialidad y la difícil clasificación del dato.

Por este motivo el tema que he seleccionado para mi Trabajo Fin de Master (TFM) es el Análisis de contenidos en Social Media y en particular problemas de agrupamiento (clustering) asociados o como coloquialmente se les llama "Clusterización en las Redes Sociales".

2. OBJETIVOS DEL PROYECTO

Este proyecto pretende avanzar en el estado del arte de la investigación acerca de los Social Media en las áreas involucradas en aspectos como la detección de regularidades en entornos de los Social Media, la creación de modelos de crecimiento, evolución y propagación o, más en general, el descubrimiento de fenómenos interesantes en torno a la dinámica de los Social Media. Estas áreas abarcarán:
La obtención de información mediante la minería del conjunto de datos procedentes de los Social Media, compuestos por contenidos, gente [datos de uso] e interacciones entre los mismos [redes sociales].
La construcción de conocimiento (modelos, métricas, etc.) a partir de la información obtenida mediante minería.

El avance en la capacidad de acción en los Social Media mediante la simulación y experimentación con el conocimiento obtenido.

3.1 Objetivo principal

• **Objetivo:**
El objetivo de carácter general de este TFM es el análisis de Social Media y el tema escogido ha sido "Banco Central Europeo". Al ser este un término tan amplio nos hemos centrado en algunos sub-problemas que se enmarcan dentro del análisis de social Media entre los que destacamos los siguientes:

1. Recogida de información de una red Social. Conectarse a una red social a través de su API y descargarnos datos relacionados con el Banco Central Europeo (BCE).
2. Depuración de datos dentro de una red social. Conectarse a diferentes foros e instituciones a través de una API creada y generada en Python, metodología de Crawling y Scraping.
3. Sentimental Analysis. Analizar el contenido y sentimiento de las menciones cuyo tema es el Banco Central Europeo (BCE).
4. Problemas de centralidad y liderazgo. Identificar a los actores más relevantes de la conversación.
5. Problemas de detección de comunidades. Identificar a las comunidades de la conversación.
6. Agregación de los tres pasos anteriores. Creación de la tabla final por individuo.
7. Problemas de Visualización de la información. Visualizar la información mediante una herramienta de visualización.

3.2 Objetivos secundarios

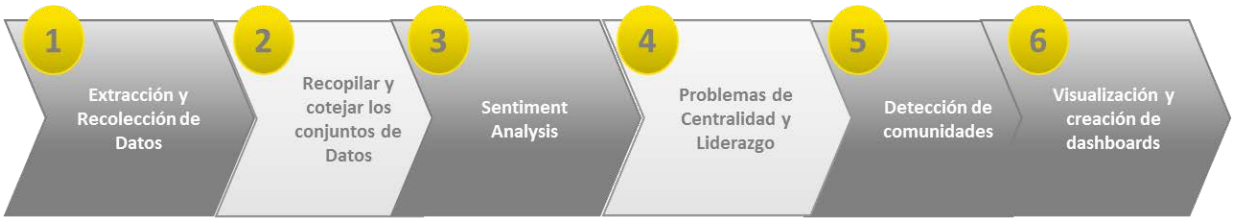
Los objetivos secundarios de este TFM, están relacionados con el plazo de entrega. Son objetivos con un enfoque de negocio.

- Proporcionar una plataforma que permita a cualquier futuro cliente controlar fuentes de noticias en torno a ciertos temas.
- Descubrir lo que la gente está discutiendo y compartiendo sobre información relacionada con el caso de uso.
- Ayuda al futuro cliente para que se anticipe a los acontecimientos, por tanto, anticipar impactos financieros, sociales, etc...
- Ayuda al futuro cliente para identificar nuevos líderes de opinión (KOL) asociados con temas relevantes.

4. METODOLOGÍA, SOFTWARE Y CRONOGRAMA

La hipótesis que se baraja es que la valoración del Banco Central Europeo (BCE) no se distribuye de forma igualitaria entre las diferentes variables categóricas de los usuarios de la Red Social Twitter, es decir, existen usuarios con mayor percepción del tema de estudio que otros.
Otra hipótesis es el estudio del comportamiento de las comunidades. A modo de ejemplo, seleccionado el Banco Central Europeo (BCE), ¿en qué se diferencia la gente que habla sobre dicha institución?, ¿hay mucha gente que habla del BCE?, ¿mucha gente se queja de las políticas económicas europeas?, es decir, la interpretación y validación de las comunidades identificadas.
Por último cabe destacar la hipótesis de la gente influyente, de la gente que habla del BCE, ¿quiénes tienen cierto peso en las redes Sociales?

4.1 Metodología de análisis

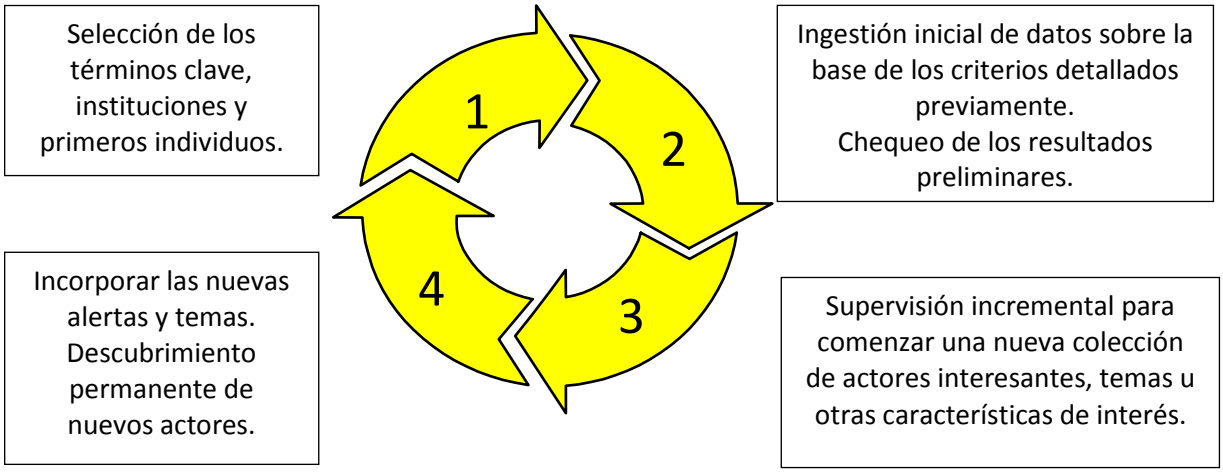


Recoger datos públicos extraído de RRSS y foros de acceso público, bases de datos disponibles, medios de comunicación social utilizando los rastreadores y aplicación del producto Interface (API).	Esta fase, se divide en la depuración de textos y la agregación de datos en torno a diferentes atributos para ser utilizados como inputs, con el fin de extraer e identificar id's únicos.	Creación de análisis de sentimiento, identificando para ello a los usuarios. Utilización de diccionarios para identificar el sentimiento de los textos.	Construcción y puesta a punto del algoritmo. Durante esta fase se define y afina nuestro algoritmo para garantizar la identificación de los actores más relevantes.	Identificar las comunidades de la conversación. Seleccionar diferentes "topics" y relacionar los actores más relevantes por comunidad.	En esta fase se visualizarán los objetivos del proyecto. Se utilizarán herramientas de Data Discovery para mostrar la evolución del proyecto, es decir, desde los análisis preliminares hasta la analítica avanzada desarrollada durante el mismo.

4.1.1 Extracción y Recolección de Datos

En esta fase se explicará con más detalle el proceso que se ha seguido para recopilar la información. Esencialmente dos fuentes han servido de punto de partida para realizar el análisis: redes sociales y blogs de internet.

El método de extracción de datos se ha producido mediante iteraciones.



4.1.1.1 Acceso a los Datos de Twitter

Twitter, un término inglés que puede traducirse como "gorjear" o "trinar", es el nombre de una red de microblogging que permite escribir y leer mensajes en Internet que no superen los 140 caracteres. Estas entradas son conocidas como tweets.
El microblogging es una variante de los blogs (las bitácoras o cuadernos digitales que nacieron como diarios personales online). Su diferencia radica en la brevedad de sus mensajes y en su facilidad de publicación (pueden enviarse desde el móvil, ordenador o dispositivos con software de mensajería instantánea).
Cuando un usuario publica un mensaje en su página de Twitter, es enviado automáticamente a todos los usuarios que hayan escogido la opción de recibirlos. Dicho mensaje también puede ser visto de forma inmediata en el perfil del usuario.

El acceso a los datos de Twitter se puede hacer por varias vías.

1. Row Data: A través de las APIs de Twitter
   - API REST
   - API Streaming
2. Curated data: A través de los proveedores oficiales de datos de Twitter (y otras redes sociales)
   - Datasift: http://datasift.com
   - GNIP (comprada por Twitter en Abril 2014) https://www.gnip.com/
   - Topsy (comprada por Apple en Diciembre de 2013) http://topsy.com/
3. Curated data + analytics: a través de otros *partners* de Twitter que venden analítica sobre los datos:
   - Brandwatch

- Hootsuite
- Mas partners en https://partners.twitter.com/

El acceso a los datos *raw* de Twitter se puede hacer a través de las dos APIs de Twitter

1. API REST (versión 1.1)
   - Permite leer tweets, escribirlos
   - Permite consultar la información de un usuario, su timeline, etc.
   - Permite buscar tweets, usuarios
   - Rate limits https://dev.twitter.com/rest/public/rate-limits
   - 180 búsquedas (queries) por cada *ventana* (15 minutos) para un usuario
   - 450 búsquedas por cada ventana para una app registrada
2. API Streaming (versión 1.1)
   - Permite recibir en tiempo real nuevos respuestas a una query tipo REST API mediante una conexión permanente
   - Útil si queremos monitorizar un evento en tiempo real o nos pasamos del límite de la REST API
   - Limits: Al número de parámetros de la query, número de resultados y solo una conexión (query) por cuenta

La extracción gratuita de información de Twitter se lleva a cabo mediante dos API's, en este trabajo se utilizará la herramienta API REST. El resultado de la consulta es un archivo en el cual puedes indicar que variables vas a seleccionar. En este caso hay variables evidentes a la hora de proceder al estudio de un tema ("topic"), como pueden ser el hashtag, nombre de usuario, fecha, descripción del tweet, retweets, etc…

4.1.1.2 Acceso a los Datos de Foros-Blogs

Web scraping se podría definir como la técnica por la que un equipo de desarrolladores es capaz de rascar, *escrapear* o liberar datos de páginas web de gobiernos, instituciones públicas u organizaciones para acceder a datos privados o públicos que puedan ser publicados o distribuidos en formato abierto.
Para realizar el proceso de Crawling y Scraping de los diferentes foros y blogs se ha utilizado Scrapy.
Scrapy es un *framework* para el rastreo y extracción de datos estructurados de páginas web. Este *framework* permite a los desarrolladores rastrear y extraer información concreta de una o varias páginas web a la vez. El mecanismo que utiliza recibe el nombre de selectores, aunque también se pueden utilizar librerías en Python como BeautifulSoup o lxml. Para este trabajo, se ha seleccionado la librería BeautifulSoup.

BeautifulSoup es una librería en Python que sirve para la extracción sencilla de datos concretos de una página web en HTML sin excesiva programación. Es lo que técnicamente recibe el nombre de *parsear* HTML. Una de las ventajas de esta biblioteca en Python es que todos los documentos salientes de la extracción de datos lo hacen en UTF-8, lo cual, es bastante interesante porque el problema típico de las codificaciones queda totalmente resuelto. Otras de las características potentes de BeautifulSoup es que utiliza analizadores de Python como lxml o html5lib, que permiten rastrear páginas web con estructura de árbol. Gracias a ellos, se puede recorrer cada 'rincón' de una web, abrirla, extraer su información e imprimirla.
Al utilizarse Python para conseguir la información de los blogs, el resultado de la consulta es un archivo de texto. En este caso no hay variables evidentes a la hora de proceder al estudio del tema, la salida es un fichero de texto plano en el cual se estudian párrafos enteros.

"ECB intervention: The ECB could purchase some of these bonds, thus ensuring long-term financing for this legacy debt at low interest rates. The risks are properly balanced: The greater the portion of these bonds purchased by the ECB, the smaller will be the burden on the domestic budgets of the member countries. Show solidarity: Countries like Greece, Ireland, Portugal and Spain are unlikely to be in a financial position to be able to deal with their debt overhang by themselves. Better-positioned countries, especially Germany, will need to make generous contributions to counter this. "

4.1.2 Recopilar y cotejar los conjuntos de Datos

[6]Es imprescindible para entender lo que es la Mineria de Textos o Text Mining, tener claro antes lo que es el Data Mining: Este último concepto surgió hace ya más de cinco años para ayudar a la comprensión de los contenidos de las bases de datos. En cualquier acto de comunicación o de tratamiento de información, de lo que se trata es

[6] http://textmining.galeon.com/

de adquirir conocimiento a partir de unos datos originales. Para el Data Mining los datos son la materia prima bruta a los que los usuarios dan un significado convirtiéndolos en información que posteriormente será tratada y utilizada por los especialistas para convertirlos en conocimiento. El data mining ha conseguido reunir las ventajas de áreas como la Estadística, la Inteligencia Artificial, la Computación Gráfica, las Bases de Datos y el Procesamiento Masivo, las bases de datos como materia prima. Molina y otros lo definirían como "la integración de un conjunto de áreas que tienen como propósito la identificación de un conocimiento obtenido a partir de las bases de datos que aporten un sesgo hacia la toma de decisión" (Molina y otros, 2001). Entendido que es el Data Mining, podemos extrapolar la misma idea a la Minería de Textos o Text mining. Los datos a tratar con esta técnica serán, en lugar de los datos tradicionales de las bases de datos, los documentos y textos de las organizaciones, administraciones, compañías, etc.

El Text Mining no se debe confundir con la recuperación de la información, que es la recuperación automática de documentos relevantes mediante indexaciones de textos, clasificación, categorización, etc. La información que realmente le interesaría a la minería de textos es aquella contenida en esos documentos pero de manera general, es decir, no está contenida en un texto en concreto sino que es la información global que tienen todos los registros, textos, documentos... de la colección en común. Es un análisis de los datos compartidos por todos los textos de la colección que se ofrece de manera indirecta, es decir, son informaciones que la colección dará a los especialistas pero que no fue específicamente incluida en esa colección en el momento de su creación para su posterior difusión a los usuarios.
Por tanto, podemos decir que la Minería de Textos comprende tres actividades fundamentales:
Recuperación de información, es decir, seleccionar los textos pertinentes.
Extracción de la información incluida en esos textos: hechos, acontecimientos, datos clave, relaciones entre ellos, etc.
Por último se realizaría lo que antes definíamos como minería de datos para encontrar asociaciones entre esos datos claves previamente extraídos de entre los textos

En resumen, la minería de textos puede ayudar a que la información implícita en los documentos más explícitos, que le ahorra tiempo y dinero. El text mining se apoya en otras técnicas como: *Categorización de texto*, *Procesamiento de lenguaje natural*, *Extracción y recuperación de la información* y finalmente en técnicas de *aprendizaje automático*.

Básicamente el procesado de lenguaje natural (NLP en inglés) nos ayuda a transformar un documento no estructurado (en este caso texto) en información estructurada. Por ejemplo:

- Detección de entidades en el texto ("voy al **Santander** a sacar dinero")
- Traducción de textos
- Análisis de sentimiento
- Reconocimiento del habla
- Desambiguación del significado de palabras
- Detección de temas, etc.

La mayoría de los algoritmos modernos de NLP están basados en técnicas de *machine learning* en los que se entrena un algoritmo para cada una de esas tareas utilizado *cuerpos* muy grandes, normalmente anotados a mano.
Para ello, se determinan una serie de variables o características (*features)* del texto, como la frecuencia de términos (bag-of-words) en el documento, la ocurrencia de n-gramas, signos de puntuación, etc. para entrenar un algoritmo de clasificación.

4.1.3 Análisis de Sentimiento

[7]El análisis de sentimiento, también conocido como minería de opinión (opinion mining), es un término muy discutido pero a menudo incomprendido.

Básicamente, es el proceso de determinar el tono emocional que hay detrás de una serie de palabras, y se utiliza para intentar entender las actitudes, opiniones y emociones expresadas en una mención online.
El análisis de sentimiento es extremadamente útil en la monitorización de las redes sociales ya que permite hacernos una idea de la opinión pública general sobre ciertos temas.

[8]En la última década, el análisis de sentimientos (SA, *sentiment analysis*), ha despertado un creciente interés. Resulta un gran reto para las tecnologías del lenguaje, ya que obtener buenos resultados es mucho más difícil de lo que muchos creen. La tarea de clasificar automáticamente un texto escrito en un lenguaje natural en un sentimiento positivo o negativo, opinión o subjetividad *(Pang and Lee, 2008)*, es a veces tan complicada que incluso es difícil poner de acuerdo a diferentes expertos humanos sobre la tarea de asignar a un texto dado un sentimiento. La interpretación personal de un individuo es diferente de la de los demás, y además se ve afectada por factores culturales y experiencias propias de cada persona. Y la tarea es aún más difícil cuanto más corto sea el texto, y peor escrito esté, como es el caso de los mensajes en redes sociales como Twitter o Facebook.

En la literatura existen, esencialmente dos enfoques para abordar este problema *(Liu, 2012)*: técnicas de aprendizaje computacional *(Pang, Lee, and Vaithyanathan, 2002)* y aproximaciones semánticas *(Turney, 2002)*.

Los **enfoques semánticos** se caracterizan por el uso de diccionarios de términos (*lexicons*) con orientación semántica de polaridad u opinión. Típicamente los sistemas preprocesan el texto y lo dividen en palabras, con la apropiada eliminación de las palabras de parada y una normalización lingüística por stemming o lematización, y luego comprueban la aparición de los términos del lexicon para asignar el valor de polaridad del texto mediante la suma del valores de polaridad de los términos. Típicamente los sistemas además incluyen un tratamiento más o menos avanzado de

[7] https://www.brandwatch.com/es/2015/02/analisis-de-sentimiento/
[8] https://www.meaningcloud.com/es/blog/introduccion-al-analisis-de-sentimientos-mineria-de-opinion

a) términos modificadores (como *muy*, *poco*, *demasiado*) que aumentan o reducen la polaridad del o los términos a los que acompañan, y b) términos inversores o negadores (como *no*, *tampoco*), que invierten la polaridad de los términos a los que afectan.
Por otra parte, los enfoques basados en **aprendizaje computacional** consisten en entrenar un clasificador usando un algoritmo de aprendizaje supervisado a partir de una colección de textos anotados, donde cada texto habitualmente se representa con un vector de palabras (bag of words), n-gramas o skip-grams, en combinación con otro tipo de características semánticas que intentan modelar la estructura sintáctica de las frases, la intensificación, la negación, la subjetividad o la ironía. Los sistemas utilizan diversas técnicas, aunque las más populares son los clasificadores basados en SVM (Support Vector Machines), Naive Bayes y KNN (K-Nearest Neighbor). En las investigaciones más recientes se han empezado a utilizar otras técnicas más avanzadas, como LSA (Latent Semantic Analisis) e incluso Deep Learning.

Análisis de sentimiento de tweets utilizando R:
Como segundo paso, se van a aplicar diferentes paquetes para conocer la opinión sobre una marca en redes sociales utilizando R. Para ello se mostrará cómo conectarnos desde R a Twitter. Buscaremos tweets asociados al Banco Central Europeo, se clasificarán las emociones y polaridad de los tweets, se visualizará la evolución de la opinión y las nubes de palabras, se detectará las opiniones negativas y a qué se deben.

4.1.4 Problemas de Centralidad y Liderazgo

La idea de centralidad aplicada a la comunicación humana fue introducida por BAVELAS en 1948. A él le interesaba en particular la comunicación en los grupos pequeños, con ese fin, creo la hipótesis sobre la relación entre centralidad estructural e influencia en los procesos grupales. Pero, ¿cómo se mide la influencia en redes sociales?
Hasta hace poco, la influencia en redes de gran tamaño se medía utilizando el **número de seguidores** o amigos en Twitter o Facebook, o el número de favoritos o likes que hacían otros usuarios en nuestros post. Hoy se utilizan otras medidas más complejas como:
Reach (alcance): el número de personas a los que se podría llegar.

Engagement: Esta medida explica y calcula la cuantía de seguidores de una persona, cuantas veces han compartido sus posts, etc. Por ejemplo, Sean Golliher descubrió (haciendo ingeniería inversa) que el número de seguidores explicaba el 95% de la varianza del índice de Klout (El índice de Klout o Klout Score es una medida de la influencia de una persona o una marca en internet.)

[9]Para los autores, las redes sociales tienen dos aspectos fundamentales: hay conexión, lo cual tiene que ver con quién está conectado con quien, y contagio, lo que fluye en la red social. Así, señalan cuatro reglas fundamentales:

[9] http://robertocarreras.com/que-es-la-influencia-en-las-redes-sociales/

Somos nosotros quienes damos forma a nuestra red. Los seres humanos organizan y reorganizan redes sociales continuamente. Como ejemplo, la homofilia, la tendencia consciente o inconsciente a asociarnos con personas parecidas a nosotros. También elegimos la estructura de nuestras redes:

- Decidimos a cuántas personas estamos conectados.
- Modificamos la forma en que nuestra familia y nuestros amigos están conectados.
- Controlamos en qué lugar de la red social nos encontramos: hacia el centro o hacia los márgenes (espero que no haya dudas del funcionamiento de una Red Social según la Teoría de los Grafos, en caso contrario os recomiendo analizar la entrada de la Wikipedia sobre redes sociales).
- Nuestra red nos da forma a nosotros.
- Nuestros amigos nos influyen. La forma de la red que nos rodea no es lo único que importan, aquello que fluye por las conexiones también es crucial. Una de las cosas que más determinan el flujo es la tendencia de los seres humanos a influenciarse y a copiarse entre sí.
- Los amigos de los amigos de nuestros amigos también nos influyen. Si quisiéramos transmitir a un grupo de personas que tiene que dejar de fumar, no las pondríamos en fila y le pediríamos a la primera que dejase de fumar y que pasase el mensaje. Por el contrario, pediríamos a muchas personas que no fuman que rodeasen a un fumador. Esta es la base real de toda la verdadera influencia que ejercen determinadas personas en los servicios de redes sociales en Internet: en ocasiones no es volumen o un medio con difusión lo que dibuja la influencia, sino que son los contactos dentro de una persona dentro de esa red en su conjunto los que hacen que un determinado mensaje tenga éxito, un contenido fluya, un producto logre difundirse en la red…
- La red tiene vida propia. Las redes sociales pueden tener propiedades y funciones que sus miembros ni controlan ni perciben. Para comprender estas propiedades hay que estudiar al grupo entero y su estructura y no sólo a individuos aislados

4.1.5 Detección de Comunidades

En este trabajo ha resultado de gran interés poner énfasis en la detección de comunidades. Esta cuestión, podría ser considerada como un problema de clustering en el que el objetivo es agrupar conjuntos de datos (que pueden ser individuos, tweets, páginas web, etc) según su parecido. No obstante, el término “detección de comunidades” difiere esencialmente de un problema de agrupamiento clásico (clustering) en que la agrupación se obtiene teniendo en cuenta las relaciones que existen entre los objetos a clasificar y no sus características o variables individuales. Dada una red se dice que tiene estructura de comunidades si los nodos pueden ser agrupados de manera natural en grupos de nodos a los que llamamos comunidades. En términos generales la idea que subyace a esta definición está basada en el principio de que dos nodos tienen mayor probabilidad de estar conectado si ambos son miembros de la misma comunidad, y menor probabilidad en otro caso.

Para este trabajo y con el objeto de detectar comunidades construiremos en primer lugar el grafo de relaciones entre individuos (grafo social), se estudiarán los enlaces que representan relaciones entre ellos con el fin de crear comunidades que tengan las siguientes propiedades.

A. Propiedades estructurales de las redes

Gran heterogeneidad
Mundo pequeño
Clustering
Comunidades
Centralidad
Homofilia

B. Propiedades de algunos procesos que suceden en ellas:

Formación de enlaces
Contagio social

Como se ha dicho antes, este problema está claramente relacionado con el clustering e intenta determinar las regiones de una red que son más densas en términos de la conducta de enlaces (*clusters* específicos por sus relaciones). El tópico está relacionado con el problema genérico de la partición del grafo, que particiona la red en regiones densas basadas en el comportamiento del enlace. Sin embargo, habitualmente las redes sociales son dinámicas y esto conduce a algunos temas únicos desde el punto de vista de la detección de la comunidad. En tales casos, el contenido se puede aprovechar en ordenar a determinar grupos de actores con intereses similares. Se han creado una serie de algoritmos importantes sobre el problema de la detección de comunidad en redes sociales de larga escala. También es factible para la investigación de detección de comunidades en medios sociales.[10]

4.1.6 Visualización y Creación de Dashboards

La visualización es un parte esencial en el Análisis de datos para una mejor comprensión de los mismos. El caso que nos ocupa no puede ser distinto de otros problemas de estadística por lo que es necesario el desarrollo de herramientas para sintetizar toda la información extraída de las Redes Sociales y Blogs de Internet. La visualización de datos no es otra cosa que el *"diseño de la comprensión"*.

Cuando se trata de hacer una visualización de muchos datos las cosas no son tan sencillas. A parte de analizar los datos, saber interpretarlos, contrastarlos con otros datos y estudiarlos, hay que saber comunicarlos. Y sobre todo es fundamental poner los datos dentro de un contexto y compararlos con algo.

[10] http://fernandosantamaria.com/blog/tag/analisis-de-redes-sociales/

Para hacer la tarea de visualización más sencilla y asequible se ha decidido reunir las mejores herramientas de visualización de datos enfocadas a Redes Sociales. Las herramientas de visualización que he elegido en general son sencillas de usar y no requieren conocimientos avanzados de desarrollo para poder utilizarlas.

La creación de visualizaciones se llevará a cabo mediante la herramienta TIBCO Spotfire. TIBCO Spotfire® Desktop es un software de análisis para la exploración de datos y la colaboración eficaz. Permite descubrir y compartir información crítica de gran valor de negocio a partir de los datos. Spotfire Desktop está diseñado para personas que desean aprovechar al máximo sus datos.

Con Spotfire se puede:

- Convertir datos en conocimiento – descubrir diferentes maneras de visualizar los datos.
- Centrarse en lo importante – resaltando las ideas más relevantes ocultas entre los datos.
- Comunicarse con precisión – compartir ideas complejas con claridad

La creación de visualizaciones enfocadas a grafos se ha desarrollado con diversas herramientas. Se ha utilizado librerías específicas de R para representar grafos, también se ha utilizado la base de datos Neo4j y por último la herramienta Gephi. A continuación se explica con más detalle la descripción de cada una de las herramientas.

4.2 Programas de análisis

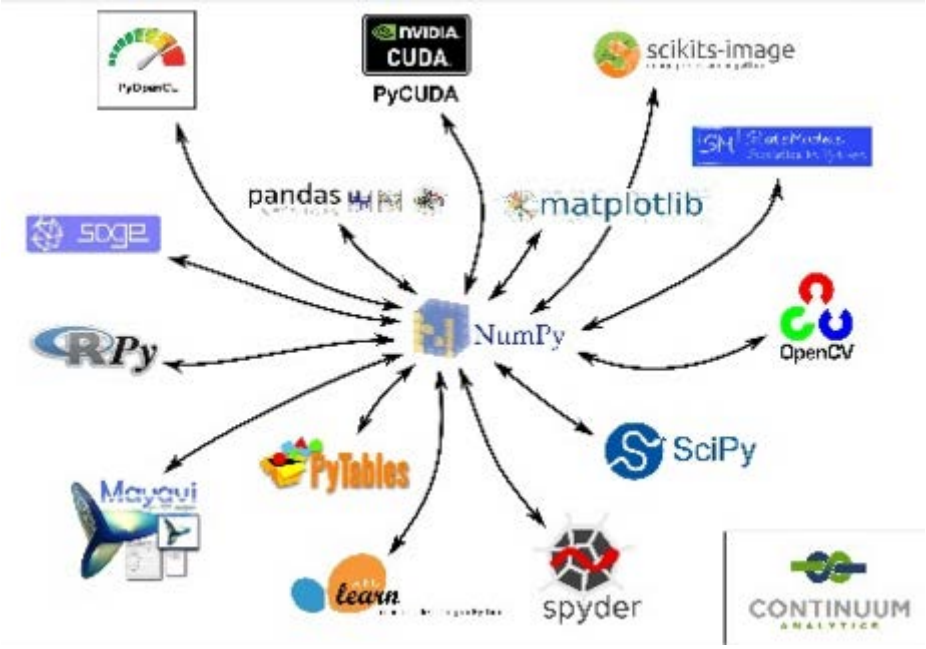
Este proyecto se realizará mediante el uso de Python, R y Spotfire. Utilizando en casos puntuales herramientas enfocadas en los grafos como pueden ser Gephi y Neo4j.

A) El programa Python se va a utilizar para descargar los datos de foros y blogs, descubrir otras fuentes de datos no estructurados.

Python es un lenguaje de programación desarrollado como proyecto de código abierto y es administrado por la empresa Python software Foundation. Fue creado por Guido van Rossum y su nombre se debe a la afición de su creador a los humoristas británicos Monty Python. Se trata de un lenguaje de programación en scripts, competencia directa de Perl. Python permite dividir el programa en módulos reutilizables desde otros programas Python. También viene con una gran colección de módulos estándar que proporcionan E/S de ficheros, llamadas al sistema, sockets, interfaces GUI, etc. Se trata de un lenguaje interpretado, lo que permite ahorrar el proceso de compilado.

Características generales de Python

- Lenguaje de programación de alto nivel del tipo scripting.
- Diseñado para ser fácil de leer y simple de implementar.
- Es código abierto (de libre uso).
- Puede ejecutarse en Mac, Windows y sistemas Unix; también ha sido portado a máquinas virtual JAVA y .NET.
- Es a menudo usado para desarrollar aplicaciones web y contenido web dinámico.
- Se utiliza para crear extensiones tipo plug-ins para programas de 2d y 3d como Autodesk Maya, GIMP, Blender, Inkscape, etc.
- Los scripts de Python tienen la extensión de archivo .PY, que pueden ser parseados y ejecutados inmediatamente.
- Permite grabar programas compilados con extensión de archivo .PYC, los cuales suelen ser usados como módulo que pueden ser referenciados por otros programas Python.
- Sitio web oficial: https://www.python.org/



B) El programa estadístico R se va a utilizar para descargar los datos de Twitter, análisis de sentimiento, descubrir a la gente influyente y crear comunidades mediante grafo.

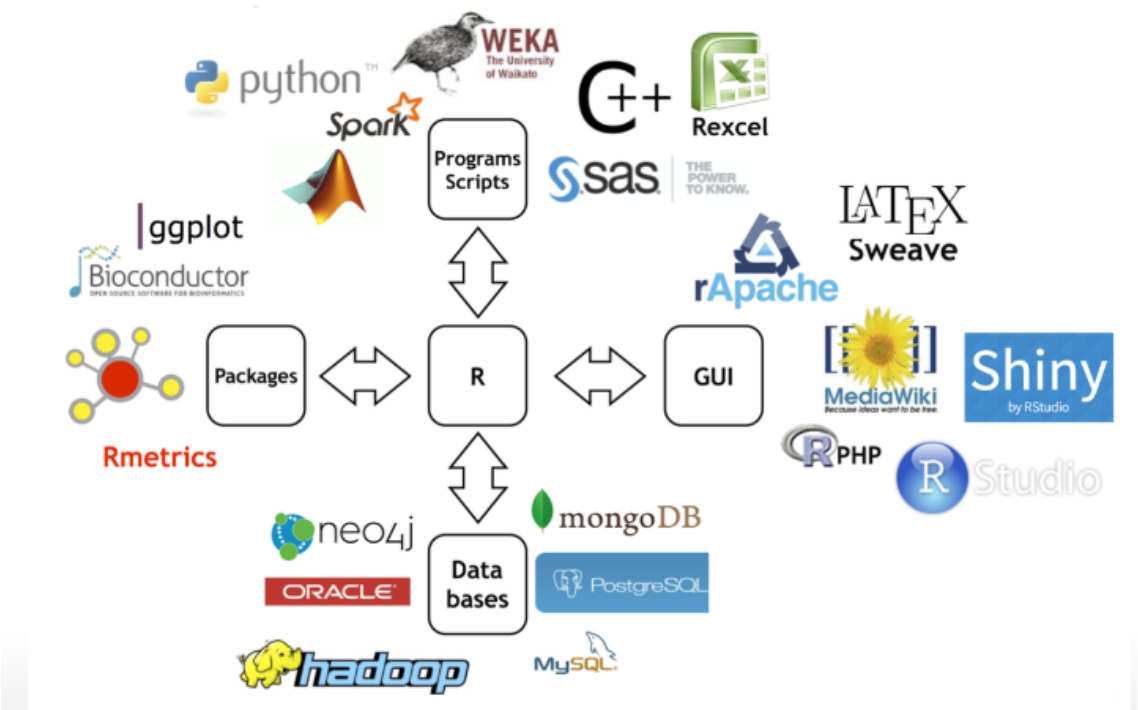
R es un potente lenguaje orientado a objetos y destinado al análisis estadístico y la representación de datos. Se trata de software libre que permite su utilización libre y gratuitamente. La comunidad científica internacional lo ha elegido como la *lingua*

franca del análisis de datos. Y tiene una gran implantación en universidades y cada vez más en mundo empresarial.

Otra definición: "*R es un paquete estadístico de última generación al mismo tiempo que un lenguaje de programación*".

Características de R:

- Es libre. Se distribuye bajo licencia GNU, lo cual significa que lo puedes utilizar y ¡mejorar!
- Es multiplataforma, hay versiones para Linux, Windows, Mac, iPhone… ¡web!
- Se puede analizar en R cualquier tipo de datos.
- Es potente. Es muy potente.
- Su capacidad gráfica difícilmente es superada por ningún otro paquete estadístico.
- Es compatible con 'todos' los formatos de datos (.csv, .xls, .sav, .sas…)
- Es ampliable, si quieres añadir algo: ¡empaquétalo!
- Hay miles de técnicas estadísticas implementadas, cada día hay más.



C) El programa Gephi se va a utilizar para moldear y representar los diferentes grafos sociales creados a través de R.

Gephi es una plataforma interactiva de código abierto (open source) para la visualización y exploración de todo tipo de redes y sistemas complejos con gráficos dinámicos y jerárquicos.

D) La BBDD Neo4j se va a utilizar para moldear y representar los diferentes grafos sociales creados a través de R.

Neo4j es una base de datos orientada a grafos escrita en Java, es decir la información se almacena de forma relacionada formando un grafo dirigido entre los nodos y las relaciones entre ellos.

E) La herramienta de BI Spotfire se va a utilizar para representar las visualizaciones de los análisis estadísticos creados a través de R.

La Plataforma Spotfire® ofrece análisis de auto-servicio. Implemente la Plataforma Spotfire® en su almacén de datos para capacitar a sus usuarios con analítica

Librerías de análisis:

Existen muchos lenguajes de programación. Por ejemplo;

Boost Graph Library (BGL) es probablemente la más conocida y antigua. Implementada en C y optimizada para ser rápida y eficiente.
SNAP (Standford Network Analysis), escrita en C++ y optimizada para grafos masivos.
NetworkX (python): paquete en python para la creación, manipulación y visualización de grafos.
[11]**Graph-tool** (python): modulo en python sobre BGL optimizada tener igual rendimiento.
[12]**igraph** (python, C y R): librería en varios lenguajes, con énfasis en la eficiencia y portabilidad.
También se han desarrollado plataformas de análisis para tratar datos masivos de redes sociales.
[13]**Giraph** (Apache): procesamiento de grafos con alta escalabilidad utilizada por Facebook, compatible con Hadoop.
Pregel (Comercial): el que utiliza google.
[14]**GraphLab** (Commercial): Toolbox con herramientas de Machine Learning para el tratamiento de grafos.

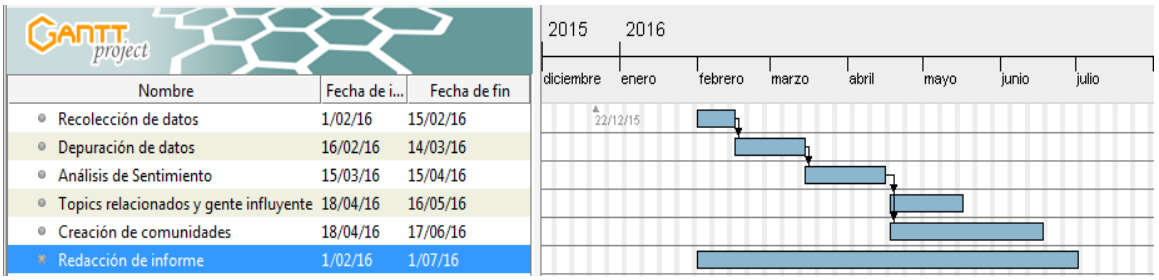
Librerías de visualización:
Existen muchas de las librerías de análisis y/o bases de datos para grafos tienen herramientas de visualización de datos.
Sin embargo también existen herramientas dedicadas a esta tarea
[15]**Gephi** es probablemente la más conocida: es una plataforma de visualización interactiva (con algunas herramientas de análisis). Funciona en Windows, Linux y MacOSX. Es el „photoshop“ de los grafos
[16]**Pajek** es un programa (para Windows) para la visualización y tratamiento de grandes grafos.
[17]**Graphviz**: librería open-source para la visualización de datos
[18]**Sigma.js** es una librería javascript para la visualización de grafos en la web
[19]**Vis.js** una librería javascript general de visualización también con aplicación a grafos
D3.js también tiene algunas visualización de grafos.
[20]**GraphX** (Apache): integrado en la solución de análisis de datos distribuida Apache Spark, es una librería de análisis de grafos y es muy fácil de utilizar.

4.3 CRONOGRAMA

Este trabajo fin de máster se ha desarrollado de acuerdo a la siguiente planificación temporal, con fecha de inicio a 1 de Febrero de 2016; momento en el que se comenzó

11 https://graph-tool.skewed.de
12 http://igraph.org
13 http://giraph.apache.org
14 https://dato.com
15 http://gephi.org
16 http://vlado.fmf.uni-lj.si/pub/networks/pajek/
17 http://www.graphviz.org
18 http://sigmajs.org
19 http://visjs.org/
20 http://spark.apache.org/graphx/

a estudiar el trabajo relacionado con las áreas propuestas. El análisis, diseño e implementación de estas actividades fue responsabilidad del alumno que ha representado el rol de Analista-Diseñador-Programador.

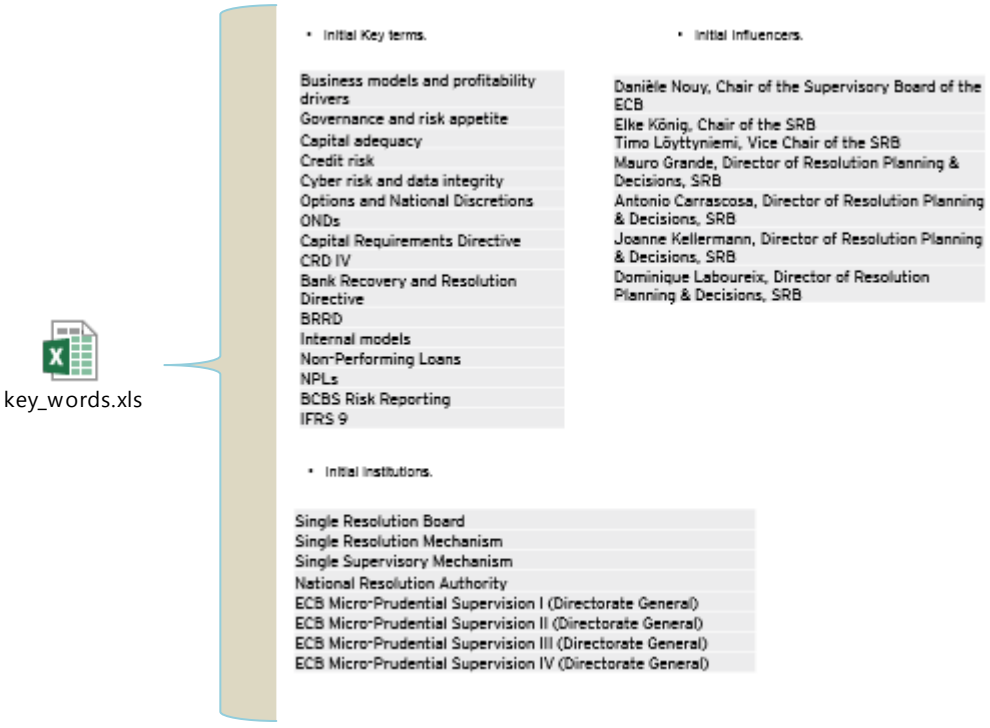


5. PROYECTO BANCO CENTRAL EUROPEO (BCE):

Una vez presentada la metodología y los objetivos fundamentales que se persiguen en este estudio, detallamos a continuación con mayor profundidad cada uno de los pasos que hemos seguido para confección de este trabajo.

5.1 Extracción y Recolección de Datos:

El proyecto se ha realizado de forma iterativa. Como primer paso se ha seleccionado una serie de términos clave relacionados con el Banco Central Europeo. La búsqueda se ha dividido por categorías que circunvalan el nodo del BCE, se ha segmentado por áreas tales como Periodistas, Políticos, Organismos, Think Tank, Profesores, Activistas….

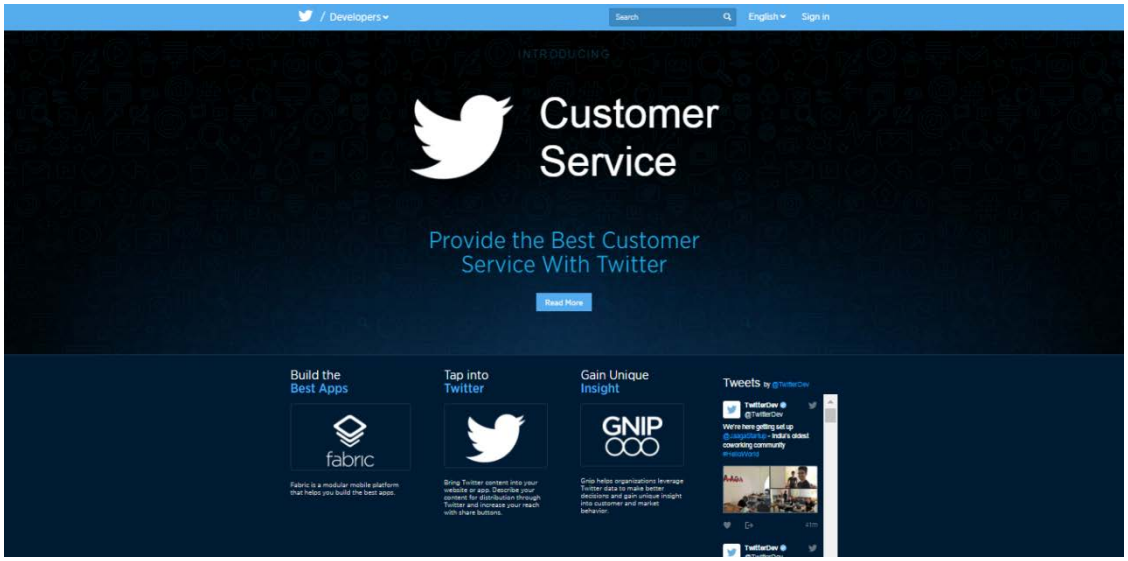


5.1.1 Twitter, utilización de las API's de Twitter y extracción de datos con el paquete estadístico R

Actualmente Twitter es una de las mayores fuentes de información en tiempo real de Internet alimentada por millones de usuarios. Twitter ofrece tres APIs:
Streaming API, REST API y Search API aplicables a necesidades diferentes.
El Streaming API proporciona un subset de tweets en casi tiempo real. Se establece una conexión permanente por usuario con los servidores de Twitter y mediante una petición http se recibe un flujo continúo de tweets en formato json. Se puede obtener una muestra aleatoria (statuses/sample), un filtrado (statuses/filter) por palabras claves o por usuarios. Sin embargo, los métodos más interesantes cómo obtener todo el caudal de tweets (statuses/firehose) o sólo los tweets que tienen enlaces (statuses/links) o los tweets con retweets (statuses/retweet).
El Search API suministra los tweets con una profundidad en el tiempo de 7 días que se ajustan a la query solicitada. Es posible filtrar por, cliente utilizado, lenguaje y localización. No requiere autenticación y los tweets se obtienen en formato json o atom. El REST API ofrece a los desarrolladores el acceso al core de los datos de Twitter. Todas las operaciones que se pueden hacer vía web son posibles realizarlas desde el API. Dependiendo de la operación requiere o no autenticación, con el mismo criterio que en el acceso web. Soporta los formatos: xml, json, rss, atom.
El Search API ofrece una información más limitada del tweet, en concreto sobre los datos del autor en el que solo indica el Id, el screen_name y la url de su avatar. Los otros dos APIs si ofrecen el perfil completo del autor en el momento de la escritura del tweet.

A modo de ejemplo, se explica la conexión y acceso a la API REST de Twitter. Como primer paso, tenemos que registrar una app con nuestro perfil en el programa de desarrolladores.

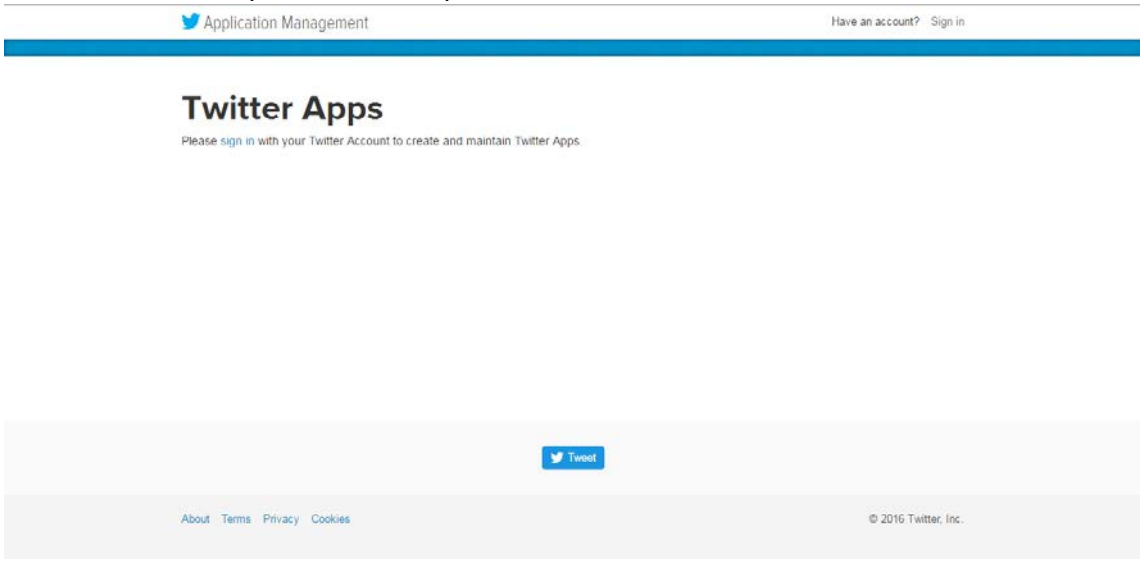
- http://dev.twitter.com



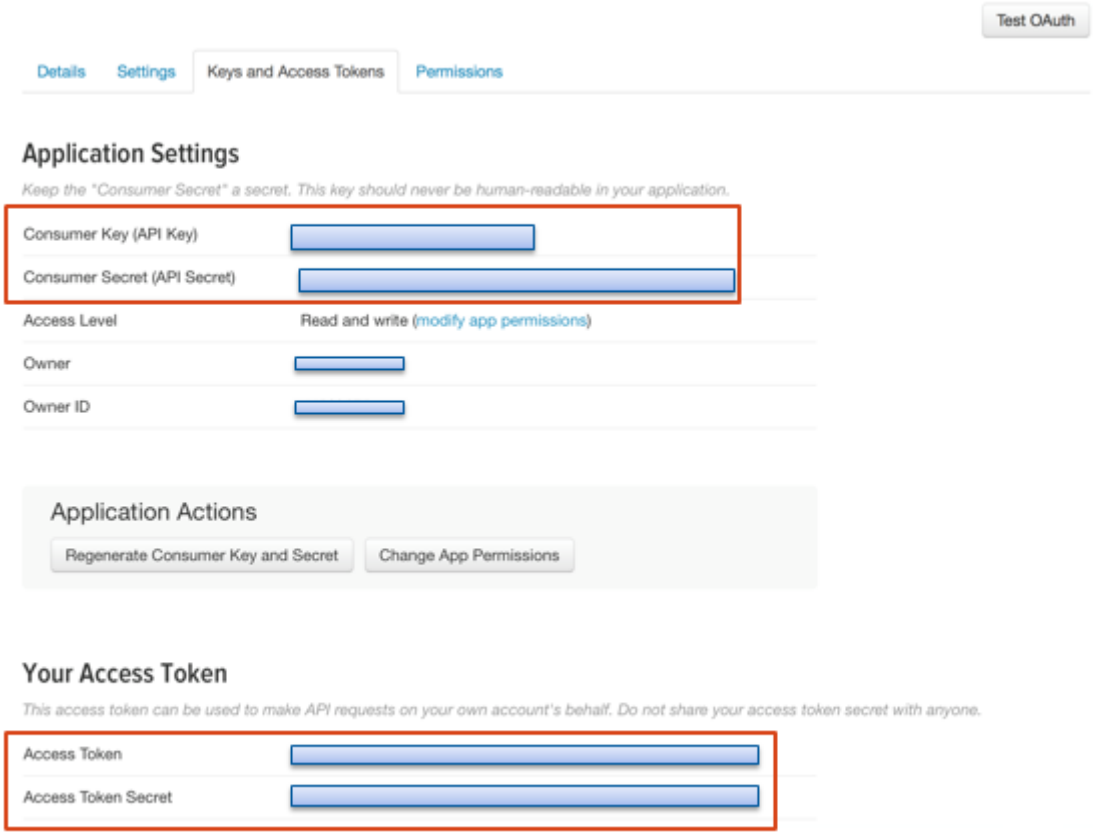
Registramos una aplicación nueva: Para conectarnos desde R a Twitter primero tenemos que registrar una aplicación en

- https://apps.twitter.com

En el Callback URL pondremos http://127.0.0.1:1410



La autenticación es mediante Oauth. Necesitamos clave de acceso a la API, es decir, API Key y API secret. Por otro lado, necesitamos las claves del token para poder acceder a nuestra aplicación (Access Token y Access Token Secret).



Finalmente para conectarnos desde R a Twitter , se procede a la conexión con Twitter. Utilizando nuestra nueva aplicación vamos a usar el paquete twitteR. Este paquete sirve para poder trabajar con todos los datos públicos que están en la API de la red social Twitter.

```
require(twitteR,quietly = T)
```

Y nos conectamos a la API utilizando nuestras claves y tokens.

```
api_key = "*************************"
api_secret = "*************************"
access_token = "*************************"
access_token_secret = "*************************"
setup_twitter_oauth(api_key,api_secret,access_token,access_token_se
cret)
```

Si todo funciona tendríamos que poder buscar en Twitter.

```
searchTwitter("ECB",n=1,lang="en")
```

Como hay límites en la API, los guardamos para tenerlos para después y luego los volveremos a cargar.

```
save(some_tweets,file="ECB.RData")
```

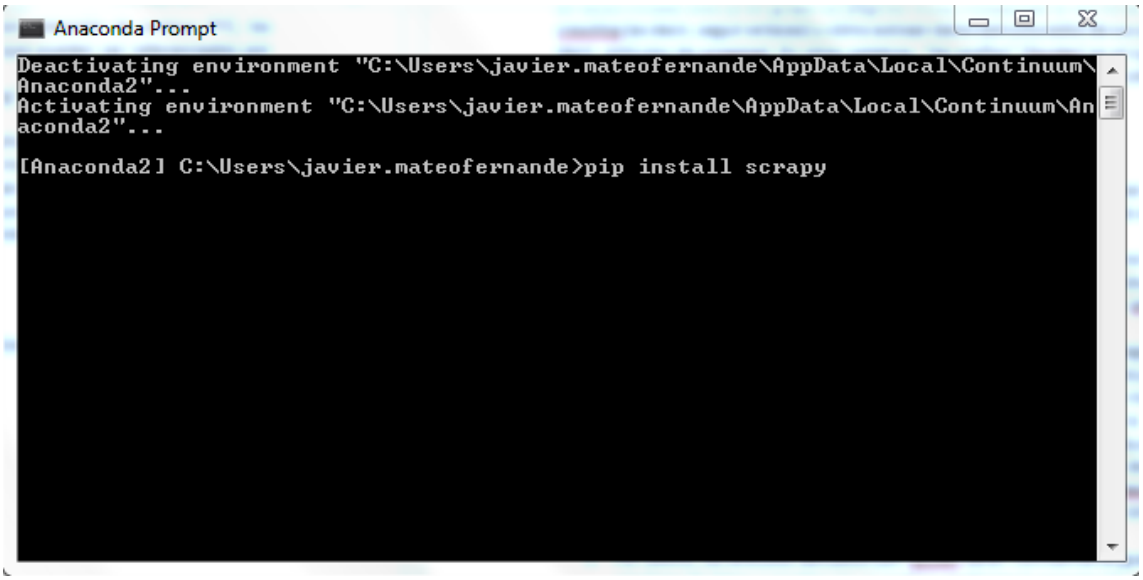
```
load("ECB.RData").
```

Una vez extraída la información de las Redes Sociales, se ha querido incrementar y enriquecer la base de datos para ello, se ha realizado la extracción de información de diferentes Blogs.

5.1.2 Foros y Blogs, utilización de la API creada en Python para la extracción de datos (Crawling&Scraping)

Para extraer la información de diversos foros y blogs del BCE se ha utilizado como herramienta de consulta Python, se ha creado una API para la búsqueda y recogida de los datos.
Como primer paso se muestra una breve explicación del programa Python.

Una vez que ya sabemos que es Python se explicará brevemente el proceso de recolección y extracción de datos públicos de esta memoria. Como primer paso hay que instalar Scrapy.
(Scrapy es un framework para python para hacer web scraping) para ello se utilizará la sentencia pip install scrapy desde el prompt de Anaconda (puede llevar unos minutos la descarga e instalación de dependencias).



Una vez instalado Scrapy podemos comenzar un proyecto nuevo con el comando:

```
scrapy startproject ECB
```

Donde *ECB* es nombre que le damos al proyecto.

Esto nos crea el siguiente árbol de directorios:

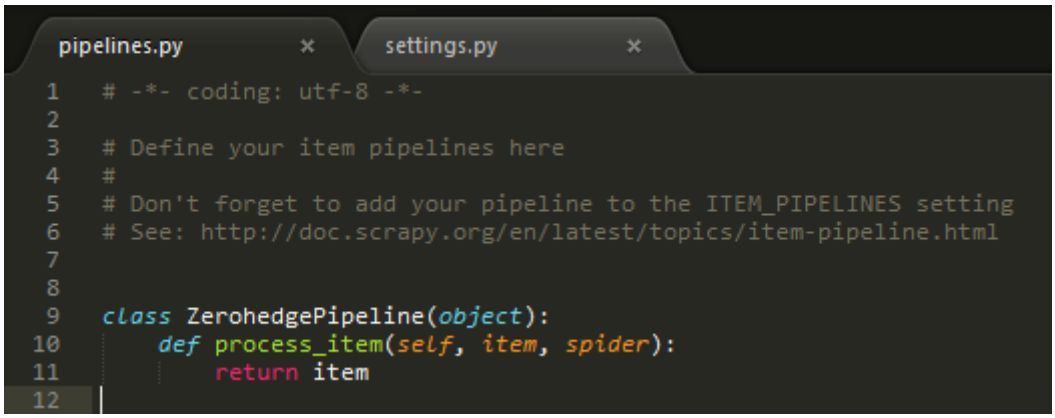
```
.
├── ECB
│   ├── __init__.py
│   ├── items.py
│   ├── pipelines.py
│   ├── settings.py
│   └── spiders
│       └── __init__.py
└── scrapy.cfg
```

Básicamente los archivos son:

- scrapy.cfg: archivo de configuración del proyecto.
- tutorial/: módulo python de nuestro proyecto, después incluiremos aquí nuestro código.
- tutorial/items.py: archivo donde definimos los items que queremos extraer.
- tutorial/pipelines.py: definimos los pipelines o flujos del proyecto.
- tutorial/settings.py: archivo de ajustes del proyecto.
- tutorial/spiders/: directorio donde luego pondremos nuestras arañas.

Una vez que ya tenemos creado el proyecto del framework Scrapy, tenemos que definir los Items del proyecto. Los *items* son contenedores que cargaremos con los

datos que extraigamos; funcionan como simples diccionarios de python pero proveen protección adicional contra campos no declarados y tipos.
Se declaran creando una clase scrapy.Item y definiendo sus atributos como objetos scrapy.Field.

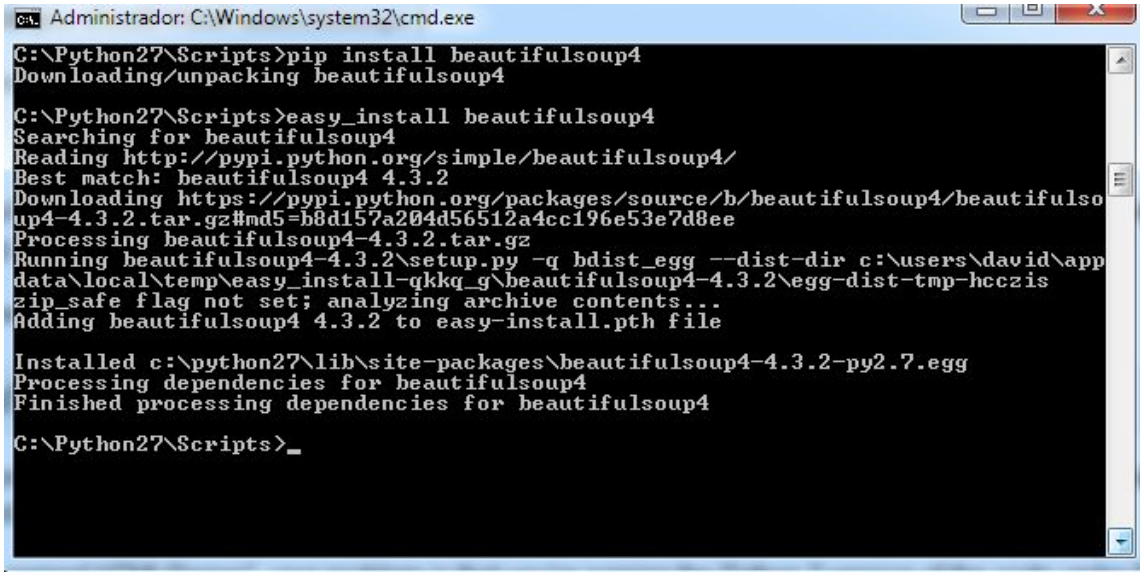


Una vez generados los Items, pasamos a configurar el siguiente módulo de paquete Scrapy. Spider son clases que definen cómo se va a realizar el scraping, seleccionando un determinado sitio (o un grupo de páginas Web), incluyendo la forma de realizar el crawling (es decir, seguir enlaces) y cómo extraer datos estructurados de sus páginas (es decir, artículos de scraping). En otras palabras, “las arañas” (Spider) es el lugar donde se define el comportamiento personalizado para el crawling y el análisis de páginas para un sitio en particular (o, en algunos casos, un grupo de sitios).

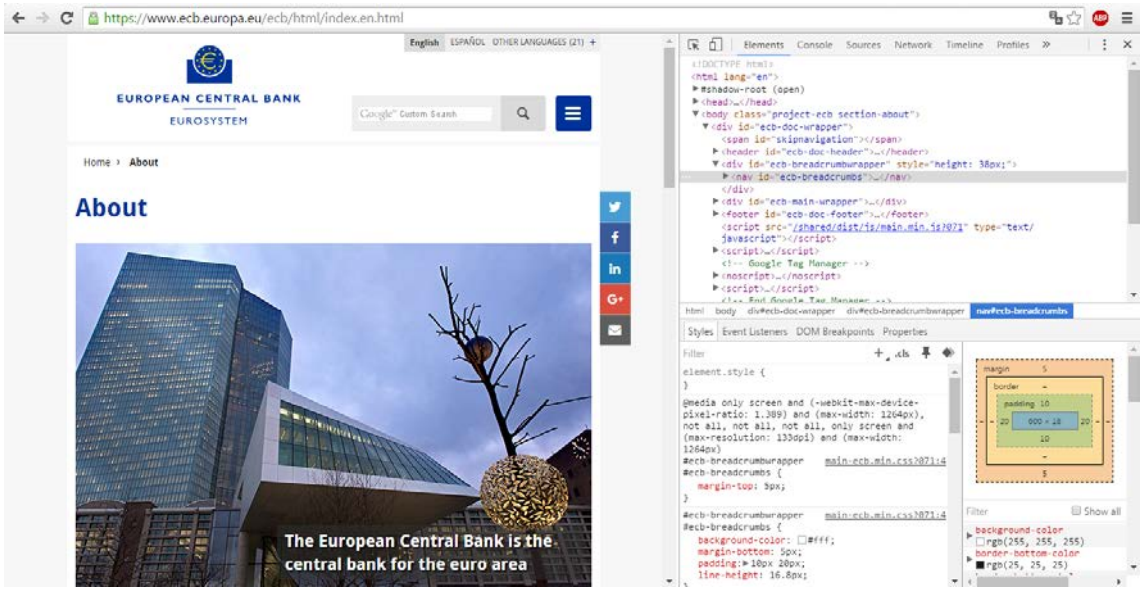
En Spider, el ciclo de scraping pasa por algo como esto:

1. Se empieza por la generación de las solicitudes iniciales para rastrear (crawling) las primeras direcciones URL, y especificar una función de devolución de llamada a ser llamados con la respuesta descargado de esas solicitudes.
   Las primeras solicitudes para llevar a cabo se obtienen mediante una llamada al método (start_requests), que (por defecto) genera una solicitud de las direcciones URL especificadas en los start_urls y el método de parse como llamada de retorno para las solicitudes.
2. En la función de devolución de llamada, se analiza la respuesta (página web) y devuelven los Items y las solicitudes con los datos extraídos del artículo. Dichas solicitudes contendrán también una devolución de llamada (tal vez la misma), y además será descargado por Scrapy, entonces se producirá la respuesta dependiendo de los parámetros introducidos en la función de devolución.
3. En las funciones de devolución de llamadas, se analiza el contenido de la página, por lo general el uso de selectores (pero también se puede utilizar BeautifulSoup, lxml o cualquier mecanismo que se prefiere) y generar elementos con los datos analizados.
4. Por último, los artículos devueltos por Spider serán normalmente guardados en una base de datos o escritos en un archivo mediante la exportación a local. Los pipelines son los encargados de guardar los datos extraídos en bases de datos, en archivos txt, en carpetas, etc.

En este proyecto se ha optado por la utilización de la librería BeautifulSoup para realizar Scraping sobre los datos. BeautifulSoup es una biblioteca de Python para la extracción de datos de los archivos HTML y XML. La librería realiza el trabajo de proporcionar formas idiomáticas de la navegación, la búsqueda y la modificación del árbol de análisis sintáctico.
A continuación se explicarán brevemente las principales características de BeautifulSoup 4. Se mostrará, cómo funciona, cómo usarlo, cómo hacer que haga lo que el usurario quiere.
En primer lugar debemos hacer un "pip install" para descargarnos e instalar los paquetes de 'request' y 'beautifulsoup'. Si no funciona “pip install beautifulsoup4” tendremos que probar con el comando “easy_install beautifulsoup4”.



En segundo lugar, se ha de estudiar la estructura del HTML de la web para ver de qué forma podemos extraer su contenido.



Una vez que se realiza el estudio del HTML y sabemos a partir de que etiquetas tenemos que scrapear los datos, vamos a pasar ahora a explicar cómo hacer el scrapeo con la librería BeautifulSoup de Python, utilizando también la librería request para hacer la petición http.

[EJEMPLO www.zerohegde.com] Es un blog antisistema muy crítico con el Banco Central Europeo.
En primer lugar definimos la url para obtener el Status Code y el HTML. En segundo lugar comprobamos el "status code", obtenemos un objeto de la clase "BeautifulSoup", pasándole el HTML de la web. Con ese objeto, vamos a poder obtener con los métodos pertinentes los elementos de la web. En este caso vamos a obtener todas las entradas que serán guardadas en una lista que corresponda a una serie de etiquetas y estilos CSS. En este caso guardaremos todos los fragmentos HTML que correspondan al tema Banco Central Europeo.

En este caso utilizamos el método "**find_all()**", lo que hace es coger todos los fragmentos del HTML que correspondan a una etiqueta div, seguido de las clases "content"'. Ahora ya tenemos en una lista (entradas), todas las entradas en HTML. Ahora debemos de recorrer esa lista y obtener de esos fragmentos de código los datos del título, autor y fecha. De la misma forma esa información está dentro de etiquetas HTML y clases CSS que debemos de tratar (como se muestra en la segunda imagen de la entrada). BeautifulSoup nos da los métodos para obtener el contenido pasándole el nombre de la clase CSS. Esto se hace de la siguiente forma:

pipelines.py | __init__.pyc | myzerohedgespider.pyc | __init__.p

```python
#-*- coding: utf-8 -*-

import re
import scrapy
from zerohedge.items import ZerohedgeItem
from scrapy.spiders import CrawlSpider, Rule
from scrapy.linkextractors import LinkExtractor

from bs4 import BeautifulSoup
#import requests

def get_post_beautifulsoup(webpage):
    page = BeautifulSoup(webpage, "html.parser")
    blog = page.find_all("div", class_ = "content")[3]
    text = blog.text.replace("\n", " ")
    text_encoded = text.encode("utf-8")
    return text_encoded

class AmazonSpider(CrawlSpider):
    regex = [re.compile("/news/")]
    otra_regex = [re.compile("/ecb")]
    #regex_more_reviews = [re.compile("/product-reviews/")]
    # bad_regex = [re.compile("/ap/forgotpassword")]

    name = "zerohedge"
    allowed_domains = ["zerohedge.com"]
    start_urls = ["http://www.zerohedge.com/search/apachesolr_search/ecb"
                  ]
    rules = [
             Rule(LinkExtractor(allow=regex),
                  follow=True,
                  callback='parse_post'
              ),
             Rule(LinkExtractor(allow=otra_regex),
                  follow=True,
                 )
              ]
```

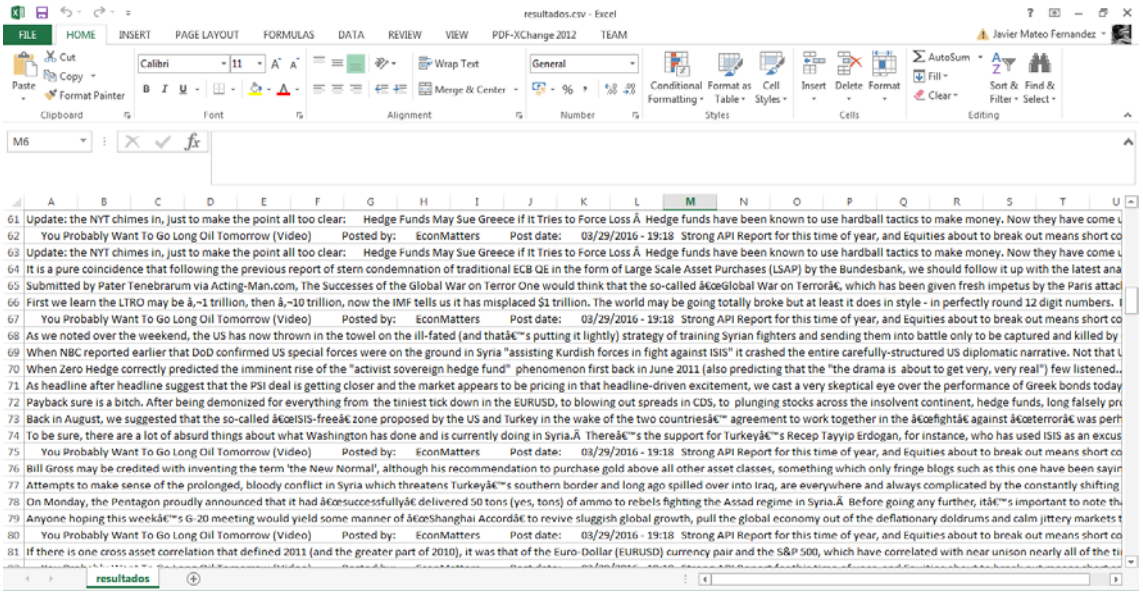
La librería BeautifulSoup, es una librería muy potente para hacer scraping. En esta entrada hemos visto cómo hacer una conexión HTTP y obtener el contenido HTML para almacenarlos en una lista para posteriormente sacar elementos o fragmentos de código HTML (**find() o find_all()**) como si de un parseo con expresiones regulares se tratase pero sin meterse en el "engorro" de trabajar con expresiones regulares. BeautifulSoup también tiene métodos para sacar el contenido de etiquetas como los h1, h2, p, img, etc. que en este caso no hemos visto en el ejemplo pero si se revisa la documentación de BeautifulSoup(https://www.crummy.com/software/BeautifulSoup/) veréis los métodos que permiten hacer esas acciones.

Para terminar dejo el programa que permite obtener todas las entradas del ECB (Banco Central Europeo) del blog zerohegde.com (título, autor y fecha) y no solo los de la página principal.

```python
import re
import scrapy
from zerohedge.items import ZerohedgeItem
from scrapy.spiders import CrawlSpider, Rule
from scrapy.linkextractors import LinkExtractor

from bs4 import BeautifulSoup
#import requests

def get_post_beautifulsoup(webpage):
    page = BeautifulSoup(webpage, "html.parser")
    blog = page.find_all("div", class_ = "content")[3]
    text = blog.text.replace("\n", " ")
    text_encoded = text.encode("utf-8")
    return text_encoded

class AmazonSpider(CrawlSpider):
    regex = [re.compile("/news/")]
    otra_regex = [re.compile("/ecb")]
    #regex_more_reviews = [re.compile("/product-reviews/")]
    # bad_regex = [re.compile("/ap/forgotpassword")]

    name = "zerohedge"
    allowed_domains = ["zerohedge.com"]
    start_urls = ["http://www.zerohedge.com/search/apachesolr_search/ecb"
                  ]
    rules = [
            Rule(LinkExtractor(allow=regex),
                 follow=True,
                 callback='parse_post'
             ),
            Rule(LinkExtractor(allow=otra_regex),
                 follow=True,
                )
             ]

    def parse_post(self, response):
        parsed_entry = ZerohedgeItem()
        response_unicode = response.body.decode(response.encoding)
        parsed_entry['entry'] = get_post_beautifulsoup(response_unicode)
        self.logger.info("%s parsed" % response.url)
        #self.logger.info("%s" % reviews['reviews'])
        return parsed_entry
```

Como punto final, seleccionamos los diferentes formatos de output y, en este caso se guardan los resultados del scraping en local.

```python
FEED_EXPORTERS_BASE = {
    'json': 'scrapy.exporters.JsonItemExporter',
    'jsonlines': 'scrapy.exporters.JsonLinesItemExporter',
    'flattenedjsonlines' : 'zerohedge.extra_exporters.JsonLinesFlattenedItemExporter',
    'csv': 'scrapy.exporters.CsvItemExporter',
    'xml': 'scrapy.exporters.XmlItemExporter',
    'marshal': 'scrapy.exporters.MarshalItemExporter',
}
```



5.2 Recopilar y cotejar los conjuntos de Datos

Para este trabajo he realizado el rastreo web, utilizando diversas fuentes de información, tales como periódicos, blogs, páginas web, etc... La finalidad en esta parte del trabajo es asociar esta información con información de las redes sociales, es decir, agregar datos de diferentes fuentes de información.
La agregación de datos es la recopilación de información de varias fuentes de datos diferentes. Mediante un proceso probado de extracción lingüística de procesamiento del lenguaje natural (NLP), se han extraído los conceptos clave de muchos tipos de datos no estructurados y agrupados en categorías. Los conceptos, opiniones y categorías extraídas se combinan con los datos estructurados con el fin de crear una única tabla de hechos.

Procesamiento de los datos recogidos:
Una vez que se recogen los datos (ya sea por la API REST de Twitter o a través de la API creada en Python), los próximos pasos implican algún tipo de pre-procesamiento, manipulación, limpieza, el formato y el filtrado.

Cabe mencionar que el total de registros recopilados asciende a 223.755, lo cual dificulta la posibilidad de mostrar los datos en completo. El análisis de los resultados no se puede plasmar en papel al necesitar gráficos interactivos.

A) Los tweets de procesamiento de datos a través de Twitter
Después de recoger los tweets deseados, necesitamos extraer su contenido. La forma más directa para extraer los datos relacionados en los tweets recogidos es mediante el uso de la twListToDF función que pone todo en una trama de datos.

```
head(ecb2$Title,2)
 [1] RT @AsiaPacNews: "No QE For You!": ECB May Cut "Lifeline" To Port
ugal After Socialists Overthrow Government https://t.co/oBs7PCaLxT #Ba
nkinâ€¦
```

```
 [2] "No QE For You!": ECB May Cut "Lifeline" To Portugal After Social
ists Overthrow Government https://t.co/I4AefdeKa7
```

El objeto ecb$Title es una lista de objetos tipo status definidos en el paquete twitteR

```
class(ecb$Title[[1]])
## [1] "status"
## attr(,"package")
## [1] "twitteR"
```

Convertimos estos objetos en una data frame.
`twListToDF:` Convierte los datos de Twitter en un data frame.

Preprocesamiento:

El preprocesador compila una colección de expresiones regulares creadas para tratar con algunas características del lenguaje, así como algunos de los elementos no gramaticales más habituales en entornos web:

- Nombres de usuarios (‘@’). En esta red social, cada usuario dispone de un alias, precedido del símbolo ‘@’ (e.g. ‘@usuarioinventado’). Sin embargo, símbolos como este pueden suponer problemas en términos de segmentación de palabras o análisis morfológico, dado que es un elemento no gramatical, característico de este medio. La estrategia del algoritmo de preprocesamiento para tratar con este fenómeno se basa en transformar los nombres de usuario a verdaderos nombres propios; eliminando el símbolo ‘@’ y convirtiendo en mayúscula la primera letra. Así, los nombres de usuarios son convertidos a nombres propios desde un punto de vista gramatical.
- Eliminación de hashtags (‘#’). Los hastags son términos incluidos en Twitter que los usuarios preceden del símbolo ‘#’, con el objetivo de etiquetar sus mensajes. Al hacer click sobre un hashtag el usuario es redireccionado al conjunto de tuits que contienen la misma etiqueta. Sin embargo, es habitual que el período de vida de los hashtags sea muy corto, dado que suelen referir eventos muy específicos (e.g. ‘#Goya2014’, ‘#SuperBowlWinner’). Por ello, este tipo de hashtags que sirven para clasificar según eventos concretos, y que son situados bien al principio o al final del tuit, son eliminados. También es frecuente utilizarlos como medio para enfatizar una palabra contenida en un mensaje (e.g ‘La #felicidad es algo difícil de conseguir’). En este caso, únicamente el símbolo ‘#’ es borrado.
- Simplificación de enlaces: En esta red social es habitual que los usuarios enlacen recursos externos, como imágenes o direcciones a otras páginas web. Con el fin de normalizar todas estas url, se utilizan expresiones regulares para detectarlas y sustituirlas por el texto ‘url’. Con ello no se persigue una normalización gramatical sino, tratar de normalizar estos elementos que pueden servir de ayuda de entrada al clasificador, como veremos en capítulos siguientes.

Para analizar el texto que hay en cada tweet tenemos que limpiarlo de

·Enlaces html
·Menciones a otros usuarios
·Retweets
·Números, signos de puntuación

```
# remove retweet entities
textos <- gsub("(RT|via)((?:\\b\\W*@\\w+)+)", "", textos) #quitar retw
eets dentro del texto
textos = gsub("@\\w+", "", textos) #quitar menciones
textos = gsub("[[:punct:]]", "", textos) #quitar signos de puntuaciÃƒÂ
³n
textos = gsub("[[:digit:]]", "", textos) #quitar nÃƒÂºmeros
textos = gsub("http\\w+", "", textos) #quitar links html
textos = gsub("[ \t]{2,}", "", textos) #quitar espacios innecesarios
textos = gsub("^\\s+|\\s+$", "", textos)
```

Observamos las diferencias una vez realizada la depuración.

```
> ecb3$Title[3]
[1] "No QE For You!": ECB May Cut "Lifeline" To Portugal After Sociali
sts Overthrow Government https://t.co/UbgjQ0gWkI
91 Levels: 'ECB moet verruimd beleid sneller afbouwen' #Financieel htt
ps://t.co/7LyiAEWzHZ ...
> textos[3]
[1] "No QE For You ECB May Cut Lifeline To Portugal After Socialists O
verthrow Government"
```

Limpiamos los textos y los pasamos a minúsculas. Para ello primero convertimos los textos al encoding correcto y nos quedamos solo con los que no son NA.

```
textos <- iconv(textos,"UTF-8","latin1")
textos <- tolower(textos)
textos <- textos[!is.na(textos)]
head(textos,3)
```

```
[1] "no qe for you ecb may cut lifeline to portugal after socialists o
verthrow government"
[2] "no qe for you ecb may cut lifeline to portugal after socialists o
verthrow government"
[3] "fedecb policy divergence will continue weighing on eurusd"
```

Una vez que tenemos los tweets limpios, vamos a determinar el sentimiento de cada uno de ellos.

5.3 Análisis de Sentimiento

El análisis de sentimiento (también conocido como minería de opinión) consiste en el uso de herramientas NLP, análisis de texto y lingüística computacional para identificar y extraer el sentimiento de un documento (actitud del autor):
La tarea básica del análisis de sentimiento es clasificar la polaridad de un texto, es decir si la opinión expresada es positiva, negativa o neutra
Más allá de esta tarea sencilla, se pueden clasificar también estados emocionales como la ira, alegría, tristeza del texto.

El análisis de sentimiento nunca es 100% preciso y depende mucho del idioma, contexto (Diccionarios), el cuerpo con el que se ha entrenado, ironía/sarcasmo ("que bien que se me haya roto el coche ahora").
Ni siquiera nos ponemos nosotros de acuerdo: numerosos estudios muestran que el porcentaje de acuerdo entre personas al determinar el sentimiento de un texto es tan sólo del 70%-80%.
El análisis de sentimiento se utiliza en muchos ámbitos. Por ejemplo:
Para conocer la opinión de los consumidores sobre productos: millones de opiniones en diferentes plataformas (tripadvisor, amazon, etc.) son clasificadas por las marcas para conocer la opinión de sus artículos
Para conocer la opinión de los ciudadanos sobre políticas: el equipo de Obama lo utilizó para conocer la opinión de los americanos antes de las elecciones del 2012.
Para estudiar la opinión política de los votantes: muchos partidos políticos (incluida en España) utilizan análisis de sentimiento para saber cuál es la opinión de sus programas, campañas, etc.
Para conocer la opinión de los inversores sobre activos/productos: muchos algoritmos de inversión en bolsa utilizan este tipo de información.

1º LIBRERIA SENTIMENT EN R:

R paquete de "sentiment":
Una opción interesante que podemos utilizar para hacer nuestro análisis de sentimientos es mediante la utilización de la del paquete sentiment de R creado por Timothy Jurka. Este paquete contiene dos funciones útiles que sirven nuestros propósitos:

Análisis de sentimiento en `R`
- Paquete `sentiment`: lamentablemente este paquete ya no está disponible, pero se puede instalar la versión antigua a través de `devtools`. Este paquete utiliza un clasificado naive Bayes entrenado sobre un léxico de emociones y otro para la subjetividad de los textos.

```
install.packages("devtools")
install.packages("Rtools")
require("devtools")
install_url("http://cran.r-project.org/src/contrib/Archive/Rstem/Rstem
_0.4-1.tar.gz")
install_url("http://cran.r-project.org/src/contrib/Archive/sentiment/s
entiment_0.2.tar.gz")
#- Paquete `syuzhet`
install.packages("syuzhet")
require(syuzhet)
#- A parte de estos paquetes tambiÃ©n se utilizan otros de procesado d
e textos como `tm`
require(tm,quietly=T)
```

#Seleccionamos como legua el inglés:

```
ecb3<-c(ecb2[ecb2$Language == "English",])

ecb3<-as.data.frame(ecb3)
```

```
textos <- ecb3$Title
```

```
library(sentiment)
```

classify_emotion

Esta función nos ayuda a analizar un texto y clasificarlo en diferentes tipos de emociones: ira, asco, miedo, alegría, tristeza y sorpresa. La clasificación se puede realizar utilizando dos algoritmos: uno es el clasificador naive Bayes entrenado por Carlo Strapparava y Alessandro Valitutti; el otro es sólo un algoritmo de votantes (simple voter algorithm).

#Obtenemos la Emoción

```
class_emo = classify_emotion(textos, algorithm="bayes", prior=1.0)
head(class_emo)
emotion = class_emo[,7] #nos quedamos el "best_fit" como emocion
emotion[is.na(emotion)] = "unknown" #sustituimos los NA por "unknown"
> head(emotion,20)
```

```
 [1] "joy"     "joy"     "unknown" "unknown" "unknown" "unknown" "unknown" "unknown"
 [9] "unknown" "unknown" "disgust" "unknown" "unknown" "joy"     "unknown" "joy"
[17] "unknown" "unknown" "unknown" "joy"
```

classify_polarity

En contraste con la clasificación de las emociones, la función classify_polarity nos permite clasificar un texto como positivo o negativo. En este caso, la clasificación se puede hacer mediante el uso de un algoritmo de naive Bayes entrenado con el léxico subjetivo de Janyce Wiebe; o mediante un simple algoritmo de votantes (simple voter algorithm).

Obtenemos la polaridad:

```
class_pol = classify_polarity(textos, algorithm="bayes")
polarity = class_pol[,4] #nos quedamos el "best_fit" como polaridad
> head(polarity,20)
```

```
 [1] "negative" "negative" "negative" "positive" "negative" "positive" "negative" "ne
utral"
 [9] "neutral"  "negative" "positive" "negative" "negative" "positive" "positive" "ne
utral"
[17] "negative" "positive" "negative" "negative"
```

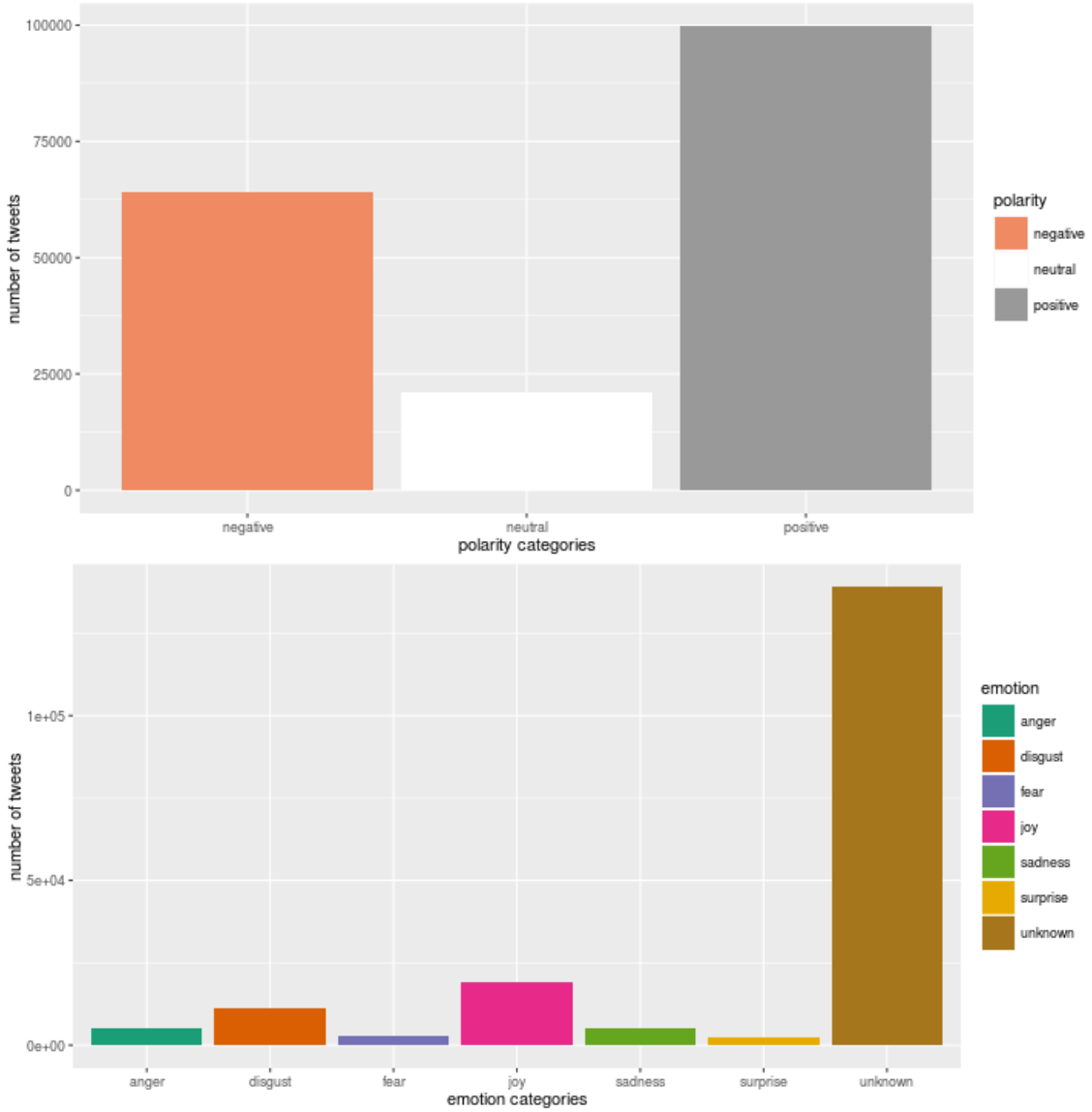
#Lo ponemos todo en una tabla

```
sent_df = data.frame(text=textos, emotion=emotion,
                     polarity=polarity, stringsAsFactors=FALSE)
tail(sent_df)
```

```
                                                                    text emotion polarity
184705                 dollar steadies as euro surge loses steam aussie trims gains unknown positive
184706 rtwhere do ray dalio see stabilitytrading finance profit best forex fx ecb gre     joy positive
184707                                        ecb defends handling of overton case unknown positive
184708                                                          bestone unknown positive
184709                                        ecb defends handling of overton case unknown positive
184710                                         boe endswith interest rates on hold     joy positive
```

#Mostramos algunas medidas de polaridad y emoción

```
plot(as.factor(sent_df$polarity))
plot(as.factor(sent_df$emotion))
```



Seleccionamos los tweets y diferenciamos por sentimiento:

```
tweets_neg <- sent_df$text[sent_df$polarity=="negative"]
tweets_neu <- sent_df$text[sent_df$polarity=="neutral"]
tweets_pos <- sent_df$text[sent_df$polarity=="positive"]
```

Utilizamos el paquete `tm` para quitar las palabras vacías (stopwords) en inglés. Y también quitamos la palabra `ECB`.

```
require(tm)
tweets_neg = removeWords(tweets_neg, stopwords("english"))
```

```
tweets_neg = removeWords(tweets_neg, c("ecb","european","bank","centra
l" ))
tweets_pos = removeWords(tweets_pos, stopwords("english"))
tweets_neu = removeWords(tweets_neu, stopwords("english"))
```
Creamos un corpus a partir de los tweets
```
corpus_neg <- Corpus(VectorSource(tweets_neg))
tdm_neg <- TermDocumentMatrix(corpus_neg)
corpus_pos <- Corpus(VectorSource(tweets_pos))
tdm_pos <- TermDocumentMatrix(corpus_pos)
```

#Nube de palabras
```
install.packages("wordcloud")
install.packages("RColorBrewer")
library(wordcloud,RColorBrewer)
wordcloud(corpus_neg, max.words=100, random.order= FALSE, colors=brewe
r.pal(8,"Dark2"),scale = c(3,.5))
wordcloud(corpus_pos, max.words=100, random.order= FALSE, colors=brewe
r.pal(8,"Dark2"),scale = c(3,.5))
```

A continuación se muestra la relación entre las palabras y las emociones en forma de wordcloud. Para ello, se ha seleccionado el texto con tweets negativos con el fin de ver la relación entre sentimiento y emoción.

```
# separating text by emotion
emos = levels(factor(sent_df$emotion))
nemo = length(emos)
emo.docs = rep("", nemo)
for (i in 1:nemo)
{
  tmp = tweets_neg[emotion == emos[i]]
  emo.docs[i] = paste(tmp, collapse=" ")
}
# remove stopwords
emo.docs = removeWords(emo.docs, stopwords("english"))
# create corpus
corpus = Corpus(VectorSource(emo.docs))
tdm = TermDocumentMatrix(corpus)
tdm = as.matrix(tdm)
colnames(tdm) = emos

# comparison word cloud
```

```
comparison.cloud(tdm, colors = brewer.pal(nemo, "Dark2"),
                 scale = c(3,.5), random.order = FALSE, title.size =
1.5)
```

A continuación se muestra la relación entre las palabras y los sentimientos en forma de wordcloud.

2º PARTE - UTILIZACIÓN DE DICCIONARIOS:

La idea general es calcular una puntuación de confianza para cada tweet para que podamos saber lo positivo o negativo que es el mensaje publicado.
Hay diferentes maneras de calcular estos índices, e incluso te puedes crear tu propia fórmula. Vamos a utilizar un enfoque muy simple pero útil para definir nuestra fórmula de calificación.

Puntuación = Número de palabras positivas - Número de palabras negativas
Si la puntuación> 0, esto significa que la sentencia tiene una "opinión positiva" en general.
Si Score <0, esto significa que la frase tiene una "opinión negativa 'global.
Si Puntuación = 0, entonces la sentencia se considera que es una "opinión neutral '.

Definimos la función score.sentiment

```
score.sentiment = function(sentences, pos.words, neg.words,
.progress='none')
{
   # Parameters
   # sentences: vector of text to score
   # pos.words: vector of words of postive sentiment
   # neg.words: vector of words of negative sentiment
   # .progress: passed to laply() to control of progress bar

   # create simple array of scores with laply
   scores = laply(sentences,
   function(sentence, pos.words, neg.words)
   {
      # remove punctuation
      sentence = gsub("[[:punct:]]", "", sentence)
      # remove control characters
      sentence = gsub("[[:cntrl:]]", "", sentence)
      # remove digits?
      sentence = gsub('\\d+', '', sentence)

      # define error handling function when trying tolower
      tryTolower = function(x)
      {
         # create missing value
         y = NA
         # tryCatch error
         try_error = tryCatch(tolower(x), error=function(e) e)
         # if not an error
         if (!inherits(try_error, "error"))
         y = tolower(x)
         # result
         return(y)
      }
      # use tryTolower with sapply
      sentence = sapply(sentence, tryTolower)

      # split sentence into words with str_split (stringr package)
      word.list = str_split(sentence, "\\s+")
      words = unlist(word.list)
```

```
      # compare words to the dictionaries of positive & negative terms
      pos.matches = match(words, pos.words)
      neg.matches = match(words, neg.words)

      # get the position of the matched term or NA
      # we just want a TRUE/FALSE
      pos.matches = !is.na(pos.matches)
      neg.matches = !is.na(neg.matches)

      # final score
      score = sum(pos.matches) - sum(neg.matches)
      return(score)
      }, pos.words, neg.words, .progress=.progress )

   # data frame with scores for each sentence
   scores.df = data.frame(text=sentences, score=scores)
   return(scores.df)
}
```

Importamos el diccionario de palabras positivas y negativas para la lengua inglesa.

```
pos = readLines("positive_words.txt")
neg = readLines("negative_words.txt")
```

Aplicamos la función score.sentiment

```
scores = score.sentiment(textos, pos, neg, .progress='text')
str(scores)
```

```
> str(scores)
'data.frame':   184710 obs. of  2 variables:
 $ text : chr  "no qe for you ecb may cut lifeline to portugal after socialists overthrow governmentbankin'
socialists overthrow government" "no qe for you ecb may cut lifeline to portugal after socialists overthrow
ighing on eurusd" ...
 $ score: int  -1 -1 -1 0 -1 0 -1 0 0 -1 ...
```

Añadimos las variables de clasificación de sentimientos al data frame

```
> scores$very.pos = as.numeric(scores$score >= 2)
> scores$very.neg = as.numeric(scores$score <= -2)
```

Vemos cuantas palabras muy positivas y cuantas muy negativas tenemos.

```
> numpos = sum(scores$very.pos)
> numneg = sum(scores$very.neg)
```

```
> numpos
[1] 7112
> numneg
[1] 11481
```

Calculamos el score global:

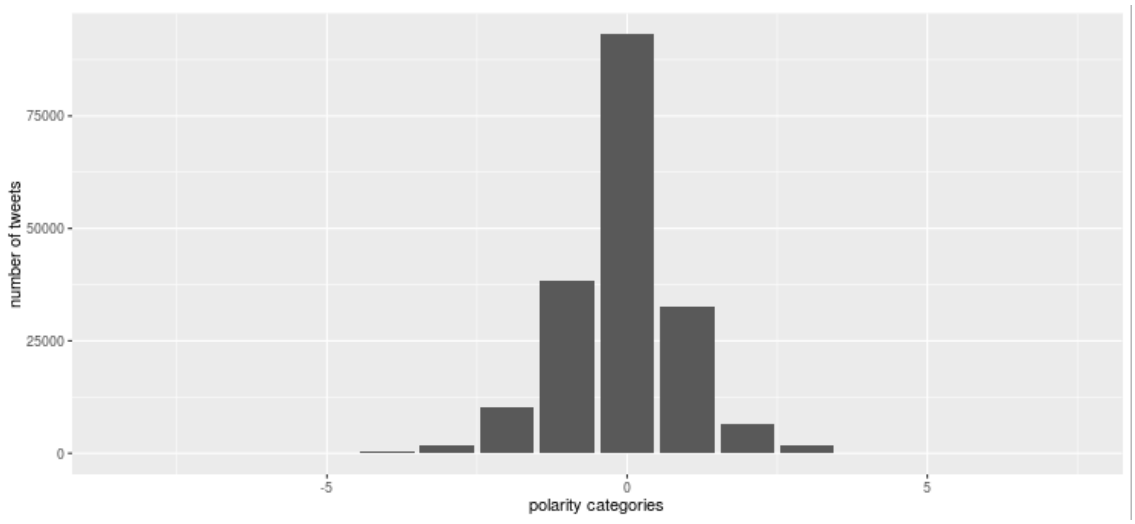
```
> global_score = round( 100 * numpos / (numpos + numneg) )
> global_score
[1] 38
```

```
head(scores)
```

	text	score	very.pos	very.neg
1	banking union ecb president mario draghi is calling for a further integration of europes banking sector	0	0	0
2	ecb constancio says monetary policy must remain accomodative	1	0	0
3	exclusive ecb mulls buying debt of cities and regionssources	-1	0	0
4	funny thingthe ecb looking at muni debt a month after madrids mayor sent the ratings agencies packing go manuela	-2	0	1
5	ecb fed ready for market jolts as they head on opposite policy paths	1	0	0
6	breakingnews ecb fed ready for market jolts as they head on opposite policy paths	1	0	0
7	sergepoznanski marketwatch banking union ecb president mario draghi is calling for a further integration of eu	0	0	0
8	crossborder markets and common governance	0	0	0
9	rehashed eu periphery enacts austerity without deleveraging until to ecbs dismay a leftwing govt is voted in	-1	0	0
10	ecb shortens transition time for some bank capital exemptionsreuters	0	0	0
11	ecb fed ready for market jolts as they head on opposite policy paths frankfurt reutersthe worlds top t	2	1	0
12	exclusive ecb mulls buying debt of cities and regionssources	-1	0	0
13	exclusive ecb mulls buying debt of cities and regionssources	-1	0	0
14	exclusive ecb mulls buying debt of cities and regionssourcesreuters	-1	0	0
15	first annual ecb macroprudential policyresearch conference jointly organised with the imf call for papers	0	0	0

Creamos un barchart con la categorización generada para el análisis de sentimiento.

```
ggplot(scores, aes(x=score)) +
  geom_bar(aes(y=..count.., fill=score)) +
  scale_fill_brewer(palette="RdGy") +
  labs(x="polarity categories", y="number of tweets")
```

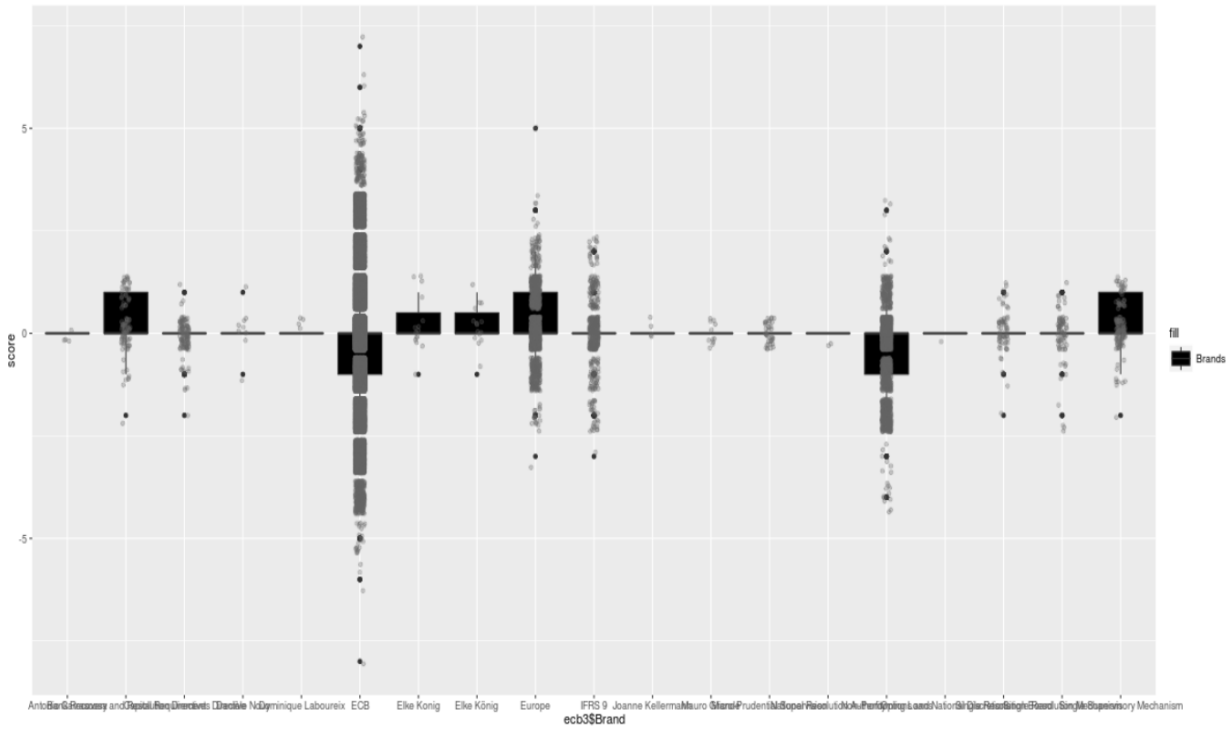


Creamos un boxplot con la categorización generada para el análisis de sentimiento. Dicho boxplot se ha creado por Brand, es decir, por los diferentes topis creados en el proceso de recolección de información.

```
# boxplot
install.packages("ggplot2")
library(ggplot2)
ggplot(scores, aes(x=ecb3$Brand, y=score, group=ecb3$Brand)) +
  geom_boxplot(aes(fill="Brands")) +
  scale_fill_manual(values=1) +
  geom_jitter(colour="gray40",
              position=position_jitter(width=0.2), alpha=0.3)
```

Como conclusión de la tercera parte del trabajo se muestra la relación entre diferentes categorías relacionadas con el Banco Central Europeo y el sentimiento que generan en

las Redes Sociales. Cabe destacar que en los medios sociales, los usuarios, de forma general, hablan bien de Europa pero no ocurre lo mismo cuando hablan de BCE. Como segunda conclusión se ha observado que la gente no está contenta con el corte de liquidez del BCE a la hora de realizar préstamos. En los siguientes pasos, la finalidad es descubrir quién habla bien y quién habla mal respecto al tema de estudio.



5.4 Problemas de Centralidad y Liderazgo

5.4.1 Introducción

Análisis de influencia en Redes Sociales:

Otra de las características del Social Media es que no todos los usuarios tienen la misma importancia a la hora de influenciar en la opinión de los demás. Esto tiene importancia en marketing ya que si convencemos a estas personas en una campaña directa, estos pueden convencer a sus amigos / seguidores. Otras veces detectar a los influenciadores nos sirve para determinar a aquellas personas que han sido más importantes en una conversación, un tema o un evento. La influencia de una persona se puede deber a muchos factores:
Porque es la que más crea el contenido
Porque es la que más lo redifunde
Porque es la que más habla de ella
Porque es la que más conexiones sociales tiene

El conocimiento del papel que juega un usuario en la adopción de un producto, en la discusión sobre una marca o en un evento es muy importante para el campo de

investigación conocido como **Influencer marketing,** una práctica en la que se orientan las actividades de marketing hacia esas personas influenciadoras, detectar las quejas, consultas de los usuarios más influyentes, vigilar las opiniones sobre productos de los influenciadores.

Como se ha mencionado antes, en redes de gran tamaño y hasta hace poco tiempo, la influencia se medía utilizando el número de seguidores o amigos en Twitter o Facebook, o el número de favoritos o likes que hacían otros usuarios en nuestros post. Hoy se utilizan otras medidas más complicadas como Reach (alcance): el número de personas a los que se podría llegar a partir de Engagement: cuanto consigue una persona que le sigan, conversen con él, compartan sus posts, etc.

5.4.2 Teoría y Métricas

En esta sección se muestran algunas de las métricas en la que se basa el proyecto. En la literatura hay una gran cantidad de medidas propuestas para determinar la influencia que tiene un nodo en una red como son Degree, Betweenness, Closeness, Eigenvector, clustering coefficient, path analysis (accesibilidad, la reciprocidad, la transitividad y distancia), el flujo, la cohesión y la influencia, y otra información útil obtenerse por varios tipos de análisis. En particular, se han utilizado inicialmente las siguientes mediciones:

- Klout Score: Klout es un sitio web y aplicación móvil que utiliza análisis de medios sociales para clasificar a los usuarios de acuerdo a la influencia social en línea, a través de la "Klout Score", que es un valor numérico entre 1 y 100. En la determinación del marcador de usuario, Klout mide la el tamaño de la red de medios de comunicación social de un usuario y correlaciona el contenido creado para medir cómo otros usuarios interactuar con ese contenido. Un alto Klout significa un letrado digital de alto. Útil en caso de seleccionar sólo las personas "digitales".
- Cantidad de seguidores: Es el número de seguidores en Twitter.
- Cantidad de menciones: Es el recuento de la mención del tweet.
- INoverOUT: La relación entre las conexiones entrantes y salientes para una persona en nuestra red, con un factor mundial aplicado a compatibilizar el nivel de influencia entre los temas - como el papel de esta persona como contribución a toda la red.

- Degree. Esta métrica tiene por objeto detectar los nodos más importantes en la red. El grado de un nodo se define como el número de conexiones directas que tiene dicho nodo con otros actores o nodos. Un nodo con centralidad (por grado) alto actúa como un centro de la red que tiene una gran cantidad de bordes de entrada y una gran cantidad de aristas que salen.

- Degrees in: La cantidad de conexiones de entrada de una persona en nuestra red. Es decir, cuántas personas interactúan con este usuario en el tema que

estamos monitoreando. En nuestro caso: el número de veces que se menciona o un retweet en twitter.

- Degrees Out: La cantidad de conexiones salientes. Es decir, cuántas personas interactúan con este usuario en el tema que estamos monitoreando. En este caso, si este usuario está mencionando o retweeting alguien en twitter. Más grado a cabo significa que las personas que buscan información fresca en nuestra red.
- Betweenness Centrality. Es el poder que tiene un nodo para intermediar en las comunicaciones de otros individuos. También conocida como poder de intermediación. Muchas medidas se han propuesto para tratar de reflejar esta idea en base a como la información fluye a través de la red. Por ejemplo, la medida de centralidad por intermediación más utilizada asume que la información en la red fluye a través de caminos mínimos. Para este caso, la centralidad por intermediación para un nodo se presenta como el número de caminos mínimos entre pares de nodos que pasan necesariamente por dicho nodo. Un nodo puede tener menos conexiones que otro nodo, pero su posición podría ser más relevante con respecto a cómo se mueve la información en la red.
- Degree Centrality. Esto es simplemente el nodo con el más alto grado (es decir, el mayor número de conexiones). En una red social, esta es la persona muy bien conectado, y la importancia de esta persona es que él o ella probablemente sabrá lo que está pasando alrededor de él, porque él o ella está tan bien conectado.
- Closeness Centrality. Aunque muchas medidas a este respecto se han definido, la más extendida está basada de nuevo en la idea de que la información se mueve a través de los caminos mínimos. Por este motivo, es común el uso de la distancia geodésica media entre un nodo y todos los demás nodos alcanzables de ella. Closeness Centrality puede ser considerada como una medida de cuánto tiempo va a tomar la información de propagarse de un nodo dado a otros nodos en la red. En una red social, esto podría ser un VIP para el que todas las comunicaciones pasan a través de unos intermediarios (por ejemplo, un asistente o un cónyuge), sino que actúa como un puente entre diferentes grupos.
- Densidad. Esta métrica se describe como nivel general de vinculación entre los nodos de un grafo. Un grafo completo es un gráfico que tiene todos sus nodos conectados directamente, es decir, cada nodo está conectado entre sí por un enlace directo. Esta medida tiene como objetivo medir qué tan lejos está el nodo del fin de la red.
- Agrupación-Segmentación. Un grupo es un sub-conjunto de nodos en los que todos los posibles pares de nodos están conectados directamente. La detección de grupos en una gráfica es importante a fin de obtener sub-comunidades.

La influencia de propagación normalmente se modela utilizando modelos de propagación tales como Linear Threshold Model y Independent Cascade Model. Estos modelos asumen que un nodo está influenciado sobre la base de las opiniones del entorno de red local. Se ha demostrado recientemente que es más simple y realista modelo la propagación de influencia negativa, es más contagioso que el modelado de

la influencia positiva. Por otra parte, basándose en la pertenencia a la comunidad para estudiar la influencia de maximización es una alternativa viable como solución.

La evaluación comunitaria y descubrimiento forman parte de los objetivos. Esto es especialmente cierto para el desarrollo social del análisis de red. En las redes sociales, una "comunidad" puede referirse a varias estructuras posibles. La definición más simple de la comunidad, como hemos visto, podría deberse a que la red tiene conexiones explícitas entre los usuarios de un servicio (amigos, seguidores, etc.). En escalas de tiempo pequeñas, estas conexiones son más o menos estáticas.
Para formar una imagen más dinámica de la estructura de la comunidad, puede ser que en lugar de determinar las comunidades en función de quién está hablando con quién o que los usuarios hablan de temas similares. De manera más abstracta, podríamos incluso definir comunidades como grupos de personas que exhiben perfiles de actividad similares. Podemos caracterizar estos tipos de comunidades en base a las preguntas que los motivan:

1. Estructura basada en: ¿Cuáles son sus amigos establecidos? ¿A quién sigue?
2. Actividad basada en: ¿Quién comparte perfiles de actividad similares?
3. Temas (Topics) basados en: ¿Acerca de qué se quiere hablar?
4. Interacción basada en: ¿Con quién se comunica?

Esto no pretende ser una lista exhaustiva, sino más bien una lista de algunos de los más tipos más comunes de comunidades observadas y estudiadas en las redes sociales.

Nos preguntamos desde el punto de vista práctico quién ha sido la persona más importante en una conversación que tuvo lugar en Twitter. Para ello vamos a utilizar los tweets que tuvieron lugar durante los meses de noviembre y diciembre del año 2015 y enfocados en el ECB.

¿Tiene alguna estructura el grafo de retweets? Al hacer un RT de una persona en un contexto como el político significa que estamos (en cierto modo) compartiendo la opinión de esa persona. Esperamos por tanto que tengamos muchos RTs dentro de una comunidad con la misma opinión sobre la conversación y pocos hacia afuera.
Vamos a estudiar por tanto las comunidades en el grafo de RT. Primero nos quedamos con la componente conexa más grande y simplificamos el grafo.

En R existen varias opciones para poder estudiar redes. Nosotros utilizaremos el paquete igraph http://igraph.org/
·Es un paquete libre para crear y manipular grafos
·Implementa los algoritmos más recientes y eficientes
·Soporta diferentes formatos de grafos
·Tiene implementaciones en C, python y por supuesto R
·Tiene posibilidades de visualización de los grafos.

También estudiaremos después el paquete visNetwork que permite una visualización interactiva de grafos http://dataknowledge.github.io/visNetwork/

·Está basado en la libreria vis.js
·Basado en htmlwidgets
·Funciona en RStudio y en los navegadores

Un grafo se dice que tiene una estructura de comunidades si los nodos se pueden agrupar en comunidades de manera que la mayoría de enlaces está dentro de esas comunidades y hay muy pocos enlaces entre comunidades.
Hay muchas maneras de calcular comunidades y diferentes criterios dan diferentes particiones del grafo. Una idea importante es la de la modularidad de una partición, que mide la bondad de la misma. En igraph tenemos
·Algoritmos basados en clustering jerárquico -*edge.betweenness* quitando enlaces de alta betweenness
-*fast.greedy* une nodos/grupos optimizando localmente la modularidad
-*multilevel.community* optimización jeráquica de la modularidad al unir nodos/grupos

·Algoritmos matriciales -*leading.eigenvector.community* utiliza el autovector principal de la matriz de modularidad

·Algoritmos basados en procesos -*label.propagation.community* basado en la búsqueda de consenso local en la vecindad de un nodo
-*walktrap.community* basado en que caminos aleatorios sobre un grafo tienden a quedarse en una misma comunidad
-*infomap.community* encuentra la estructura de comunidades que minimiza la descripción de las trayectorias de caminantes aleatorios en el grafo.

Comenzamos estudiando el campo del tweet donde tenemos la información del usuario. Como se puede observar, necesitamos depurar el fichero puesto que la extracción de información no ha podido relacionar todos los nombres de usuario.

```
head(ecb3$User)
[1] YuukiFushimi    frederic_san11  tamaraspen2     MrtenStrmberg2  robertdavis4811 f
xsignals4pips
55778 Levels:  -=?????????????=- ?? ??-???? ??? ??? ?? ???? ??? ???? ?? ???? ???? ???
????? ... 麦芒资本
```

```
influence<-ecb3$User
# remove retweet entities
influence   <-   gsub("(RT|via)((?:\\b\\W*@\\w+)+)",   "",   influence)
#quitar retweets dentro del texto
influence = gsub("@\\w+", "", influence) #quitar menciones
influence  =  gsub("[[:punct:]]",  "",  influence)  #quitar  signos  de
puntuaciÃ³n
influence = gsub("[[:digit:]]", "", influence) #quitar nÃºmeros
influence = gsub("http\\w+", "", influence) #quitar links html
influence  =  gsub("[  \t]{2,}",  "",  influence)  #quitar  espacios
innecesarios
influence = gsub("^\\s+|\\s+$", "", influence)
```

#miramos la diferencia

```
> ecb3$User[3]
```

```
[1] tamaraspen2
55778 Levels:  -=????????????=- ?? ??-???? ??? ??? ?? ???? ??? ???? ?? ???? ???? ???
????? ... 麦芒资本
> influence[3]
[1] "tamaraspen2"
```

Construimos tres medidas de influencia:

·Actividad: número de tweets por usuario

·Retweets: número de RTs recibido y centralidad en el grafo del RTs

Creamos primero la tabla de actividad (table crea una tabla de contingencia)

```
actividad <- data.frame(table(tolower(influence)))
colnames(actividad) <- c("user","ntweets")
head(actividad)
```

```
> head(actividad)
                                          user ntweets
1                                                 8725
2 자나깨나완벽법해법전략전술꿈사랑즐기는법유희장       1
3                                            a      58
4                                       aaaaaa       1
5                                      aaaaace       1
6                                        aaafx       3
```

Ordenamos los usuarios por número de Tweets. Cabe destacar la existencia de un montante de 8725 tweets sin usuario, eliminamos dichos tweets de la base de datos.

```
head(actividad[order(actividad$ntweets, decreasing = T),])
```

```
              user ntweets
1                    8725
14156   forexwarrior     799
13604  finanzlinksus     754
14859  fxsignalspips     592
27468     moneynewsh     571
4882  blackcentaurfx     523
```

Seleccionamos aquellos tweets que son RTs e identificamos el usuario que crea y recibe el RT.

```
> rts <- grep("^.*rt @[a-z0-9_]{1,15}", tolower(ecb3$Title), perl=T)
>     rt.sender     <-     gsub("^.*@([a-z0-9_]{1,15})+.*$",     "\\1",
tolower(influence[rts]), perl=T)
>    rt.receiver    <-    gsub("^.*rt    @([a-z0-9_]{1,15})+.*$",    "\\1",
tolower(ecb3$Title[rts]), perl=T)
> head(rts) #NOS DICE POR POSICION QUE TWEETS SON RETWETTS
[1]  1  4  7 12 16 21
```

Creamos el grafo de RTs. CADA NODO ES UN USUARIO Y CADA EDGE ES UN RETWEET.

```
install.packages("igraph")
> require(igraph,quietly=T,warn.conflicts = F)
> edgelist <- data.frame(rt.sender,rt.receiver,stringsAsFactors=F)
> str(edgelist)
'data.frame':     43997 obs. of  2 variables:
```

```
 $ rt.sender  : chr  "yuukifushimi" "mrtenstrmberg" "elibaram" "primemarketssa" ...
 $ rt.receiver: chr  "asiapacnews" "jbjakobsen" "zerohedge" "bloombergtv" ...
```

```
> g <- graph.data.frame(edgelist)
> g
```

```
IGRAPH DN-- 23448 43997 --
+ attr: name (v/c)
+ edges (vertex names):
 [1] yuukifushimi  ->asiapacnews    mrtenstrmberg ->jbjakobsen     elibaram      ->zerohedge
 [4] primemarketssa->bloombergtv    cfdseducation ->livesquawk     ragham        ->zerohedge
 [7] gaganianev    ->zerohedge      openewseu     ->zerohedge      nordeamarkets ->jbjakobsen
[10] fedupusa      ->zerohedge      feldart       ->breakingviews  damian        ->zerohedge
[13] huuractiebreda->went1955       tonykelly     ->zerohedge      forextrail    ->fxstreetnews
[16] twittyshoes   ->zerohedge      peedeeheenee  ->zerohedge      simmimi       ->zerohedge
[19] menaforexexpo ->dukascopyfx    lessismoremike->zerohedge      zerohedgert   ->zerohedge
[22] coryhasiak    ->zerohedge      joseavsantos  ->richardcalhoun brawnm       ->ecb
+ ... omitted several edges
```

TENEMOS 23448 USUARIOS Y 43997 RETWEETS

Nos quedamos en el grafo solo con aquellos que han hecho más de un RT.

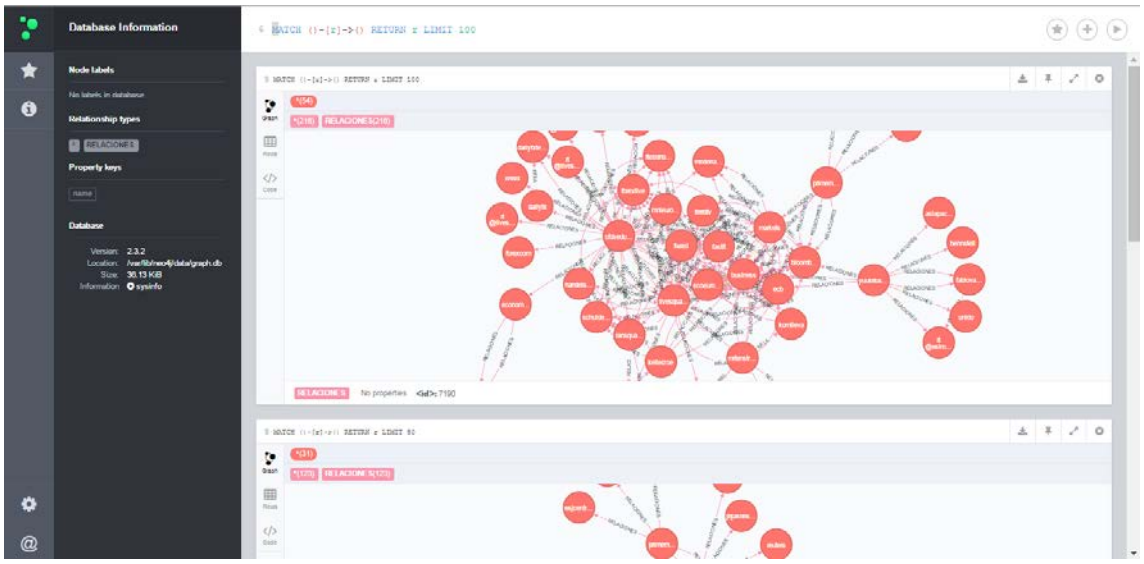
```
> g0 <- induced_subgraph(g,which(degree(g)>1))
> g0
```

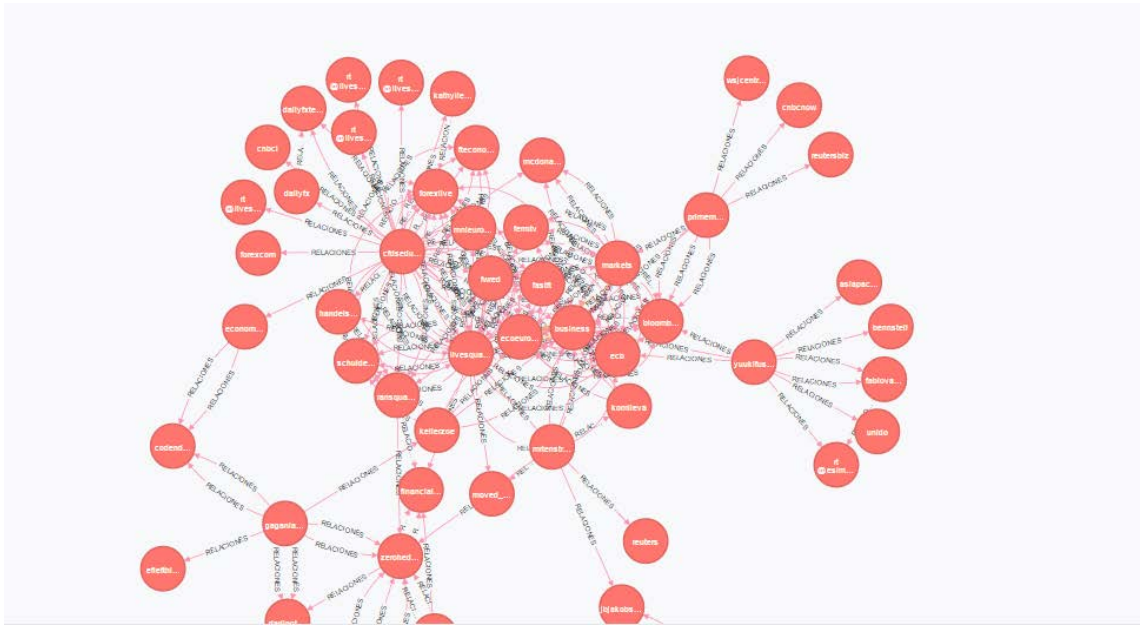
```
> g0
IGRAPH DN-- 7850 29250 --
+ attr: name (v/c)
+ edges (vertex names):
[1] yuukifushimi->fabiovanorio yuukifushimi->fabiovanorio yuukifushimi->business
[4] yuukifushimi->ecb          yuukifushimi->bloombergtv  yuukifushimi->bennsteil
[7] yuukifushimi->asiapacnews
+ ... omitted several edges
```

TENEMOS 7850 USUARIOS Y 29250 RETWEETS

```
write.graph(g0,file="g2_bis.graphml",format="graphml")
```

Una vez guardamos en grafo social, se ha estudiado en la base de datos Neo4j. Se lanzan peticiones para obtener de forma visual el tipo de relación entre nodos.





Calculamos la centralidad en este grafo como una aproximación del **reach**

```
degRT <- degree(g0,mode = "in")#numero de retweets
```

```
> head(degRT)
  yuukifushimi  mrtenstrmberg       elibaram primemarketssa  cfdseducation     gaganianev
             0              0              0              0              0              0
> str(degRT)
 Named num [1:7850] 0 0 0 0 0 0 0 3 0 0 ...
 - attr(*, "names")= chr [1:7850] "yuukifushimi" "mrtenstrmberg" "elibaram" "primemarkets
```

```
cenRT <- page_rank(g0)$vector # centralidad del page rank
```

```
> str(cenRT)
 Named num [1:7850] 5.51e-05 5.51e-05 5.51e-05 5.51e-05 5.51e-05 ...
 - attr(*, "names")= chr [1:7850] "yuukifushimi" "mrtenstrmberg" "elibaram"
> head(cenRT)
  yuukifushimi  mrtenstrmberg       elibaram primemarketssa  cfdseducation
  5.509585e-05   5.509585e-05   5.509585e-05   5.509585e-05   5.509585e-05
```

```
cenRT_ben <- betweenness(g0) #centralidad utilizando betweeness
str(cenRT_ben)
head(cenRT_ben)
```

```
> str(cenRT_ben)
 Named num [1:7850] 0 0 0 0 0 ...
 - attr(*, "names")= chr [1:7850] "yuukifushimi" "mrtenstrmberg" "elibaram"
> head(cenRT_ben)
  yuukifushimi  mrtenstrmberg       elibaram primemarketssa  cfdseducation
             0              0              0              0              0
```

Formamos la tabla de influencia

```
influenciaRT <- data.frame(user=V(g0)$name,degRT,cenRT,cenRT_ben)
head(influenciaRT)
```

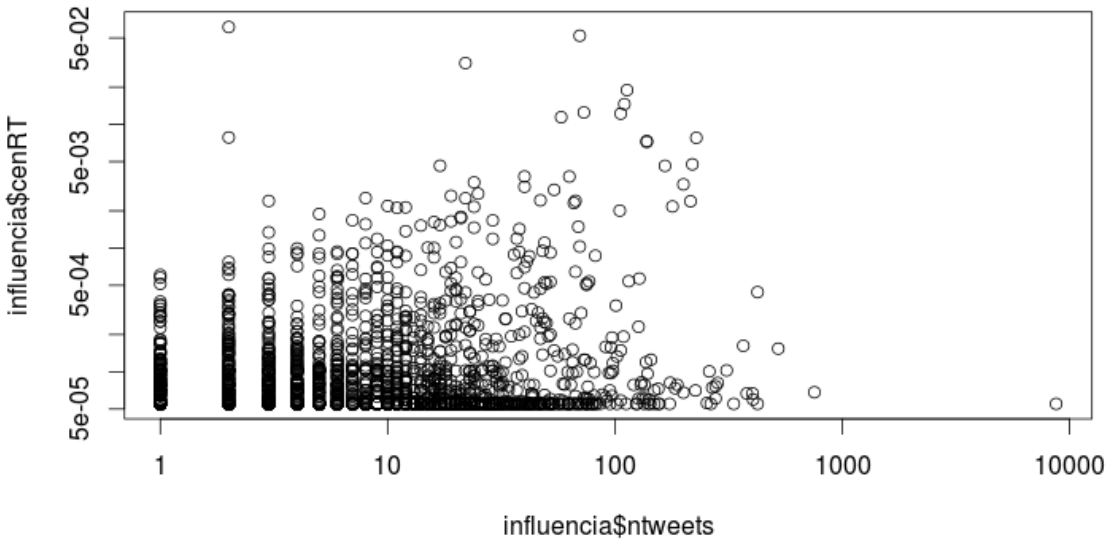
```
> head(influenciaRT)
                       user degRT        cenRT cenRT_ben
yuukifushimi     yuukifushimi     0 5.509585e-05         0
mrtenstrmberg   mrtenstrmberg     0 5.509585e-05         0
elibaram             elibaram     0 5.509585e-05         0
primemarketssa primemarketssa     0 5.509585e-05         0
cfdseducation   cfdseducation     0 5.509585e-05         0
gaganianev         gaganianev     0 5.509585e-05         0
```

Juntamos las dos tablas

```
influencia <- merge(actividad,influenciaRT)
head(influencia[order(influencia$cenRT,decreasing=T),])
```

```
> head(influencia[order(influencia$cenRT,decreasing=T),])
            user ntweets degRT      cenRT  cenRT_ben
5617   scarletfu       2    15 0.06129621      0.000
929     business      70  2082 0.05210868 230674.718
796  bloombergtv      22   119 0.03132003   6253.253
4025     markets     113   364 0.01894620 311134.479
1771         ecb     110  1391 0.01458431      0.000
6890   zerohedge      73  1027 0.01250182  10847.348
```

Como se observa en el gráfico inferior, una gran actividad en la conversación no garantiza una gran centralidad.



El objetivo de este proyecto es responder a una serie de preguntas como;

¿Quiénes son los usuarios más influyentes?

La respuesta es compleja, depende de la métrica que queramos mostrar tendremos gente que es más influyente que otros.

```
> head(influencia[order(influencia$cenRT,decreasing=T),],3)
            user ntweets degRT      cenRT  cenRT_ben
5617   scarletfu       2    15 0.06129621      0.000
929     business      70  2082 0.05210868 230674.718
796  bloombergtv      22   119 0.03132003   6253.253
```

Si contamos el número de tweets como medida de influencia, cabe la pena mostrar los 8725 tweets sin usuario, si nos finamos en el gráfico anterior observamos que es un dato atípico.

```
> head(influencia[order(influencia$ntweets,decreasing=T),],4)
               user ntweets degRT        cenRT cenRT_ben
1                    8725     0 5.509585e-05    0.0000
2146  finanzlinksus     754    13 6.834877e-05  591.5252
783  blackcentaurfx     523    23 1.540290e-04    0.0000
935  businessnewzzz     424    67 4.407526e-04    0.0000

> head(influencia[order(influencia$cenRT_ben,decreasing=T),],3)
          user ntweets degRT       cenRT cenRT_ben
2341      fwred     138   375 0.007334340  401706.8
3781 livesquawk     200   507 0.003274669  353538.4
4025    markets     113   364 0.018946204  311134.5

> head(influencia[order(influencia$degRT,decreasing=T),],3)
                user ntweets degRT      cenRT cenRT_ben
929         business      70  2082 0.05210868  230674.7
5626 schuldensuehner     106  1535 0.01222089       0.0
1771             ecb     110  1391 0.01458431       0.0
```

¿Quiénes son los usuarios que han participado mucho pero no han tenido casi RTs?

Cabe destacar que algunos de los siguientes usuarios pueden ser Bots.

```
> head(influencia[influencia$ntweets>100 & influencia$cenRT <1,])
              user ntweets degRT        cenRT cenRT_ben
1                      8725     0 5.509585e-05       0.0
206     alerttrade     137    12 1.196772e-04      72.8
261 allinonesgnews     106     2 5.862679e-05       0.0
555   aussietorres     109     9 1.924048e-04  127271.4
618     bamabroker     102     0 5.509585e-05       0.0
773    bizdatabase     175     5 8.159001e-05       0.0
```

Una vez calculadas las diferentes métricas utilizadas para conseguir la influencia de un usuario. Se ha seleccionado a los usuarios más influyentes por categoría. En el estudio de las redes sociales existen tres categorías a priori de usuario, “Compañías”, “Media” e “Individuos” (Personas Independientes), la realizada es que no todo el mundo tiene dicha categorización de forma predeterminada, por lo tanto , se ha creado otro grupo de usuarios con la categoría “Otros”.
A continuación se muestra una tabla detalle con el top de usuarios por categoría.

1º Influyentes por Compañía:

User	Date	Hour	Influencers	User Followers Count	Impacts	Betweenness	Degrees Out	Degrees In	INoverOUT
UKIP	12/11/2015	11:57:37	Highest	111.369,00	334.125,00	122.637,90	2,00	26,00	10,38
UBS	18/11/2015	12:56:13	Highest	150.722,00	754.818,00	106.856,40	3,00	33,00	8,78
saxobank	13/11/2015	13:38:13	Highest	25.679,00	1.314.766,00	275.003,30	42,00	34,00	0,65
PIMCO	21/11/2015	13:02:28	Highest	196.149,00	2.161.751,00	330.723,50	11,00	100,00	7,26
OppFunds	03/12/2015	21:25:18	Highest	59.258,00	177.846,00	49.483,71	9,00	29,00	2,57
NASDAQ	12/11/2015	18:26:01	Highest	445.418,00	445.418,00	4.709,09	1,00	7,00	5,59
GoldmanSachs	04/12/2015	15:14:06	Highest	430.310,00	861.479,00	81.591,40	3,00	61,00	16,23
Gendarmerie	25/11/2015	15:12:28	Highest	160.401,00	160.401,00	14.864,20	2,00	157,00	62,66
FXCM	12/11/2015	8:33:26	Highest	61.041,00	2.404.848,00	98.257,58	4,00	75,00	14,97
forex_us	12/11/2015	10:09:27	Highest	217.867,00	1.082.765,00	3,17	1,00	3,00	2,39
FitchRatings	04/12/2015	16:00:36	Highest	99.586,00	199.220,00	9.153,96	2,00	12,00	4,79
EuroParlPress	12/11/2015	13:28:47	Highest	54.576,00	54.576,00	1,00	1,00	1,00	0,80
Europarl_EN	14/11/2015	18:55:05	Highest	184.805,00	184.805,00	32.614,22	1,00	22,00	17,56
DIFC	22/11/2015	11:40:07	Highest	398.420,00	398.420,00	101,53	1,00	2,00	1,60
DeutscheBank	13/11/2015	14:55:15	Highest	313.528,00	5.031.473,00	431.233,30	18,00	110,00	4,88
CMEGroup	03/12/2015	17:22:08	Highest	768.487,00	1.536.977,00	9.749,07	2,00	132,00	52,68
BofAML	30/11/2015	13:00:10	Highest	94.977,00	94.977,00	4.360,36	1,00	3,00	2,39

2º Influyentes por Media:

User	Date	Hour	Influencers	User Followers Count	Impacts	Betweenness	Degrees Out	Degrees In	INoverOUT
USATODAY	03/12/2015	22:19:51	Highest	1.912.407,00	3.827.364,00	148.299,90	2,00	91,00	36,32
TODAYonline	23/11/2015	17:30:01	Highest	304.015,00	915.846,00	650,14	2,00	3,00	1,20
timesofindia	30/11/2015	19:21:00	Highest	5.786.233,00	11.600.424,00	14.140,50	1,00	15,00	11,97
FinancialReview	25/11/2015	19:53:10	Highest	110.977,00	1.233.091,00	88.060,60	8,00	33,00	3,29
Fin24	03/12/2015	20:41:25	Highest	108.018,00	650.120,00	52.922,93	6,00	5,00	0,67
euronews	04/12/2015	1:33:27	Highest	195.182,00	390.634,00	10.702,01	2,00	9,00	3,59
dailyEEUU	12/11/2015	17:55:48	Highest	11.424,00	341.643,00	31.841,82	33,00	5,00	0,12
CNNMoney	30/11/2015	22:19:46	Highest	1.153.345,00	11.536.857,00	12.411,57	2,00	347,00	138,49
CNBC	01/12/2015	23:48:24	Highest	2.197.637,00	43.817.261,00	8.099,67	1,00	282,00	225,09
BusinessDesk	20/11/2015	6:05:22	Highest	125.708,00	2.262.901,00	25,99	1,00	37,00	29,53
bsindia	25/11/2015	12:34:22	Highest	202.929,00	1.822.192,00	9.135,47	2,00	13,00	5,19
BloombergTV	25/11/2015	1:07:02	Highest	343.056,00	7.616.122,00	104,83	1,00	280,00	223,50
BBCSussex	29/11/2015	13:28:37	Highest	50.571,00	50.571,00	9.124,00	1,00	1,00	0,80
bbcbusiness	03/12/2015	1:45:15	Highest	1.638.765,00	6.556.194,00	20,40	4,00	4,00	0,80
abcnews	23/11/2015	12:47:32	Highest	906.276,00	906.276,00	2,33	1,00	4,00	3,19
NewstalkFM	12/11/2015	12:42:20	Highest	120.698,00	1.344.695,00	35.070,79	1,00	21,00	16,76
telegraaf	10/12/2015	12:43:32	Highest	406.400,00	7.197.358,00	389.316,50	12,00	101,00	6,72

3º Influyentes por Individuo:

User	Date	Hour	Influencers	User Followers Count	Impacts	Betweenness	Degrees Out	Degrees In	INoverOUT
Thais_ecb	14/11/2015	0:35:53	High	1760	68759	4680,09	5	6	0,9578428
Tanzeel_Akhtar	03/12/2015	14:27:47	High	5193	25958	0,3333333	1	1	0,7982023
TamiHoffman	03/12/2015	11:59:12	High	2093	4185	4309	1	1	0,7982023
talorne	24/11/2015	0:07:41	High	1075	1075	2,75	1	8	6,385618
Aidan_Regan	30/11/2015	7:27:33	High-Medium	763	763	12	2	1	0,3991012
ahmannt	11/11/2015	18:07:48	High-Medium	504	2040	18264,82	4	3	0,5986517
adamsamson	04/12/2015	19:43:29	High	3381	16898	20,33333	1	7	5,587416
AdamPosen	12/11/2015	11:25:48	Highest	17748	178522	169,1675	3	26	6,917753
adam_tooze	10/12/2015	19:28:05	High	1242	10505	8956,46	1	11	8,780225
Accendo_Mike	11/11/2015	14:15:35	High	2848	112402	51098,88	6	9	1,197303
acardenasfx	06/12/2015	20:40:58	High	8265	123394	17,8125	1	6	4,789214
ABitsch	03/12/2015	12:47:32	High-Medium	616	1852	27,59829	4	1	0,1995506
78laurablanco	03/12/2015	12:00:55	High	8156	16236	3	1	3	2,394607
777mingus	12/11/2015	12:50:00	High-Medium	543	2176	312,187	3	1	0,2660674
4xForecaster	27/11/2015	14:46:05	High	1772	19425	49406,7	15	3	0,1596405
4nnak	03/12/2015	13:21:05	Highest	33355	166780	9557,24	5	4	0,6385618
_norman_g	03/12/2015	9:00:34	High	8140	113617	15352,48	4	10	1,995506

4º Influyentes Categorizados como Otros:

User	Date	Hour	Influencers	User Followers Count	Impacts	Betweenness	Degrees Out	Degrees In	INoverOUT
TotalKaiser	30/11/2015	9:05:44	High-Medium	475,00	954,00	33.800,44	2,00	9,00	3,59
top_grafisch	03/12/2015	17:49:21	High-Medium	798,00	6.354,00	4.311,00	10,00	1,00	0,08
rekapberita	03/12/2015	23:48:13	High-Medium	163,00	326,00	3,00	2,00	2,00	0,80
redroute	08/12/2015	10:03:11	High	6.608,00	171.926,00	1,00	1,00	1,00	0,80
NT_CTannenbaum	24/11/2015	21:30:20	High-Medium	9,00	60,00	4.309,00	3,00	1,00	0,27
notyourcountry	11/12/2015	12:46:26	High-Medium	186,00	5.985,00	318.606,50	37,00	2,00	0,04
Markit	24/11/2015	11:30:06	Highest	15.130,00	45.423,00	8.648,37	3,00	5,00	1,33
KP2060	09/12/2015	12:25:03	High-Medium	540,00	9.728,00	3,50	3,00	1,00	0,27
InvestHuddle	18/11/2015	20:11:59	High-Medium	77,00	2.597,00	38.755,11	25,00	4,00	0,13
AberdeenAssetUS	03/12/2015	17:50:33	High	7.083,00	14.162,00	137,53	2,00	3,00	1,20
1DannyStewart	23/11/2015	4:57:56	High-Medium	158,00	158,00	4,00	1,00	1,00	0,80
_s_U_r_f_E_r_	04/12/2015	21:11:53	High-Medium	118,00	826,00	117,87	7,00	1,00	0,11
_FrontierMan	21/11/2015	18:11:56	High-Medium	119,00	928,00	9.182,56	4,00	1,00	0,20
_Economie	08/12/2015	8:21:27	High	1.042,00	36.244,00	5.019,52	43,00	3,00	0,06
_ChrisVersace	01/12/2015	14:39:42	High	1.895,00	7.582,00	6.159,92	5,00	5,00	0,80
_BuddhistTrader	07/12/2015	9:19:43	High	5.148,00	44.399,00	4.309,00	11,00	1,00	0,07
_adriapeiro	20/11/2015	8:18:15	High-Medium	203,00	1.039,00	0,50	1,00	1,00	0,80

5.5 Detección de Comunidades

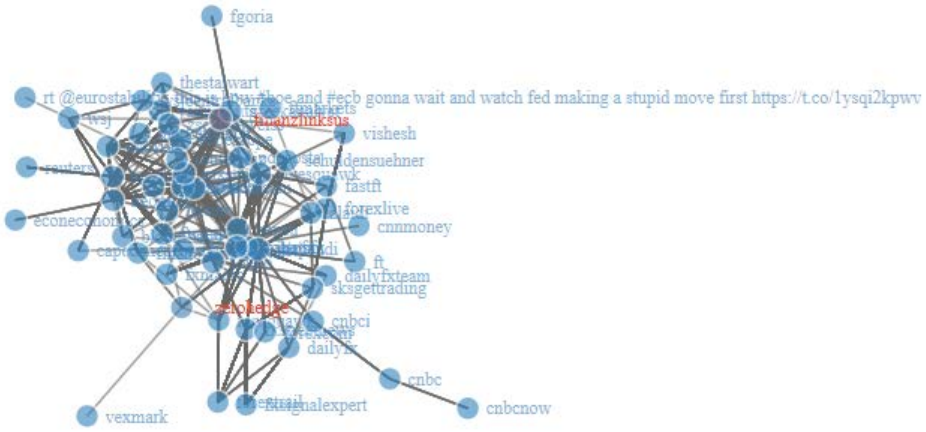
¿Tiene alguna estructura el grafo de RTs? Al hacer un RT de una persona en un contexto como el político significa que estamos (en cierto modo) compartiendo la opinión de esa persona. Esperamos por tanto que tengamos muchos RTs dentro de uan comunidad con la misma opinión sobre la conversación y pocos hacia afuera.

Vamos a estudiar por tanto las comunidades en el grafo de RT. Primero nos quedamos con la componente conexa más grande y simplificamos el grafo.

Se ha querido estudiar la comunidad de influyentes para el topic ECB. Para ello se han seleccionado las personas que tienen más de 100 nexos de unión.

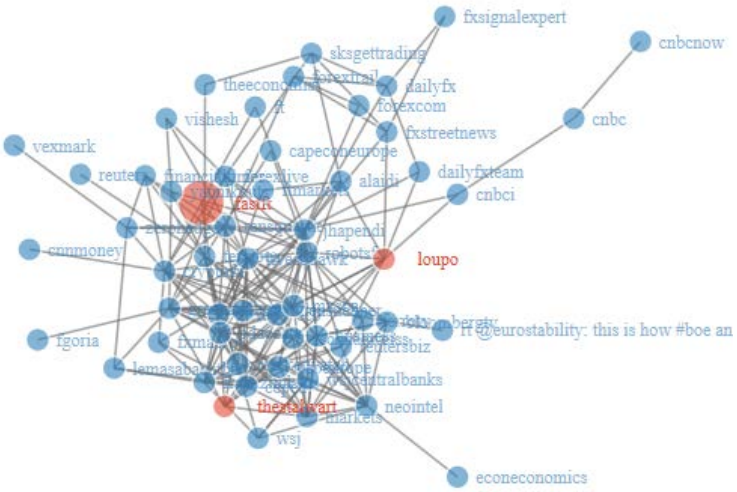
```
demo<-c(ecb[ecb$Brand == "ECB",])

influence_demo<-demo$User
# remove retweet entities
influence_demo      <-      gsub("(RT|via)((?:\\b\\W*@\\w+)+)",      "",
influence_demo) #quitar retweets dentro del texto
influence_demo = gsub("@\\w+", "", influence_demo) #quitar menciones
influence_demo   =   gsub("[[:punct:]]",   "",   influence_demo)   #quitar
signos de puntuaciÃ³n
influence_demo   =   gsub("[[:digit:]]",   "",   influence_demo)   #quitar
nÃºmeros
influence_demo  =  gsub("http\\w+",  "",  influence_demo)  #quitar  links
html
influence_demo   =   gsub("[   \t]{2,}",   "",   influence_demo)   #quitar
espacios innecesarios
influence_demo = gsub("^\\s+|\\s+$", "", influence_demo)
#Creamos primero la tabla de actividad (`table` crea una tabla de
contingencia)
actividad_demo <- data.frame(table(tolower(influence_demo)))
colnames(actividad_demo) <- c("user","ntweets")
head(actividad_demo)
head(actividad_demo[order(actividad_demo$ntweets, decreasing = T),])
#Seleccionamos aquellos tweets que son RTs e indentificamos el usuario
que crea y recibe el RT
rts <- grep("^.*rt @[a-z0-9_]{1,15}", tolower(demo$Title), perl=T)
rt.sender       <-       gsub("^.*@([a-z0-9_]{1,15})+.*$",       "\\1",
tolower(influence_demo[rts]), perl=T)
rt.receiver    <-    gsub("^.*rt    @([a-z0-9_]{1,15})+.*$",    "\\1",
tolower(demo$Title[rts]), perl=T)
head(rts) #NOS DICE POR POSICION QUE TWEETS SON RETWETTS
#[1]  1  4  7  8 10 16
#Creamos el grafo de RTs. CADA NODO ES UN USUARIO Y CADA EDGE ES UN
RETWEET
install.packages("igraph")
require(igraph,quietly=T,warn.conflicts = F)
edgelist_demo <- data.frame(rt.sender,rt.receiver,stringsAsFactors=F)
g_demo <- graph.data.frame(edgelist_demo)
g_demo


g0_demo <- induced_subgraph(g,which(degree(g)>100))
```

```
g0_demo

install.packages("networkD3")
library(networkD3)
library("igraph")

# Extract into data frame and plot
g <- get.data.frame(g, what = "edges")

g0_demo<- get.data.frame(g0_demo, what = "edges")

simpleNetwork(g0_demo, fontSize = 12)
```



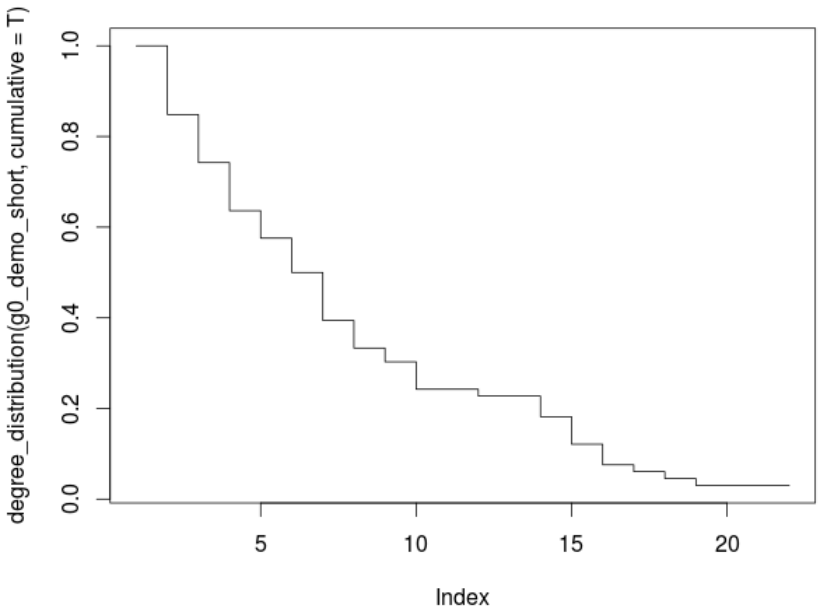
Simplificamos el grafo para quedarnos con relaciones simples entre los nodos.

```
g0_demo_short <- simplify(g0_demo)
g0_demo_short<- get.data.frame(g0_demo_short, what = "edges")
simpleNetwork(g0_demo_short, fontSize = 12)
```



Veamos algunas propiedades del grafo. Como siempre sale un grafo muy heterogéneo.

```
plot(degree_distribution(g0_demo_short,cumulative=T),type="s")
```



También tiene las propiedades de mucha asortatividad y transitividad.

Transitividad: las redes sociales son densas localmente. La probabilidad de que dos vecinos de un nodo sean a su vez vecinos es muy alta (25%).

```
> transitivity(g0_demo_short)
[1] 0.2558775
```

Los vecinos más próximos muestran una correlación en la conectividad social. Una red asortativa es aquella en la que los nodos que están muy conectados tienden a estar rodeados de nodos con alta conectividad.

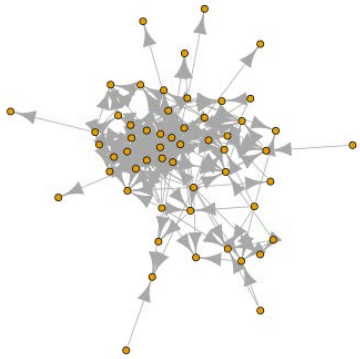
```
> assortativity_degree(g0_demo_short)
[1] 0.06831063
```

No es un grafo totalmente conectado.

```
> is.connected(g0_demo_short)
[1] FALSE
```

Vamos a visualizar como es el grafo localmente alrededor de un nodo, seleccionamos el nodo zerohedge, observando grafos anteriores sabemos que es un nodo central e influyente.

```
g0_cen <- make_ego_graph(g0_demo_short,4,6)[[1]]
plot(g0_cen,vertex.color=V(g0_cen)$zerohegde,vertex.size=4,vertex.label="")
```



Un grafo se dice que tiene una estructura de comunidades si los nodos se pueden agrupar en comunidades de manera que la mayoría de enlaces está dentro de esas comunidades y hay muy pocos enlaces entre comunidades.
Hay muchas maneras de calcular comunidades y diferentes criterios dan diferentes particiones del grafo. Una idea importante es la de la modularidad de una partición, que mide la bondad de la misma. En igraph tenemos
·Algoritmos basados en clustering jerárquico -edge.betweenness quitando enlaces de alta betweenness
-fast.greedy une nodos/grupos optimizando localmente la modularidad
-multilevel.community optimización jeráquica de la modularidad al unir nodos/grupos

·Algoritmos matriciales -leading.eigenvector.community utiliza el autovector principal de la matriz de modularidad

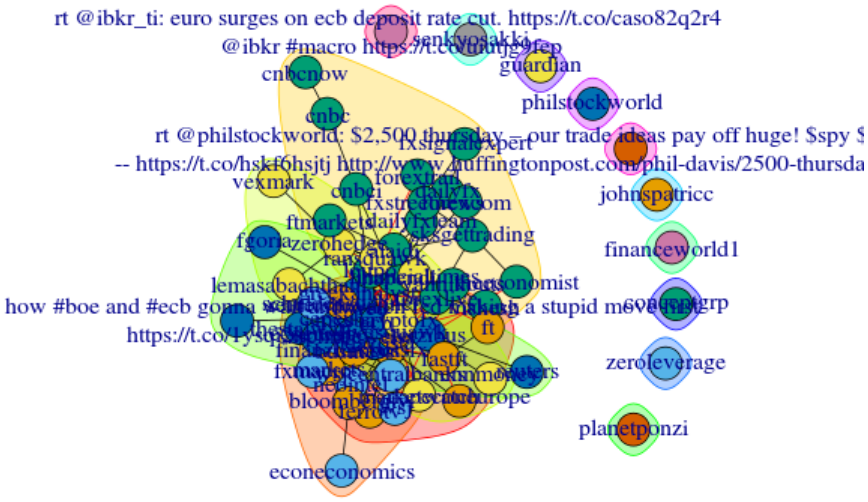
·Algoritmos basados en procesos -label.propagation.community basado en la búsqueda de consenso local en la vecindad de un nodo
-walktrap.community basado en que caminos aleatorios sobre un grafo tienden a quedarse en una misma comunidad

-infomap.community encuentra la estructura de comunidades que minimiza la descripción de las trayectorias de caminantes aleatorios en el grafo.

```
g <- as.undirected(g0_demo_short)

fg <- fastgreedy.community(g)
```

Con la siguiente bondad de la partición
```
modularity(fg)
```
[1] 0.276644

·Obtenemos los enlaces que están dentro y entre comunidades
```
crossing(fg,g)
```
·Finalmente mostramos el grafo y las comunidades
```
plot(fg,g)
```



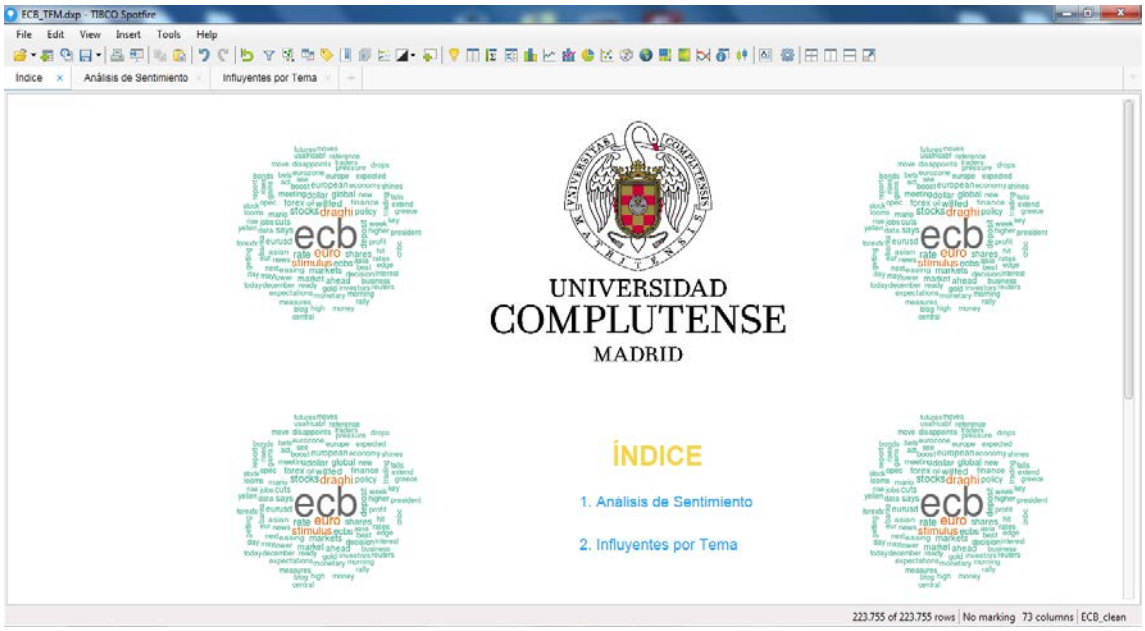
Ejemplo Ilustrativo

Finalmente se han generado 18 comunidades. La gran mayoría de los usuarios están englobados dentro de la comunidad ECB (European Central Bank) puesto que ha sido el principal motor de búsqueda de la recolección de datos. Cabe destacar que existen individuos que generan sus propias comunidades, este es el caso de “Elke Konig”, “Mauro Grande”, etc..., son pequeñas comunidades que se mueven entorno a el BCE.

Brand	Porcentaje
Bank recovery and Resolution	0,066%
Capital Requirements Directive	0,092%
ECB	97,733%
Europe	0,636%
IFRS 9	0,247%
Micro-Prudential Supervision	0,033%
National resolution Authority	0,001%
Non Perfoming Loans	0,977%
Options and National Discreptions	0,000%
Single Resolution Board	0,038%
Single Resolution Mechanism	0,046%
Single Supervision Mechanism	0,086%
Antonio Carrascosa	0,004%
Daniéle Nouy	0,013%
Dominique Laboureix	0,002%
Elke Konig	0,018%
Joanne Kellerman	0,002%
Mauro Grande	0,007%
	100%

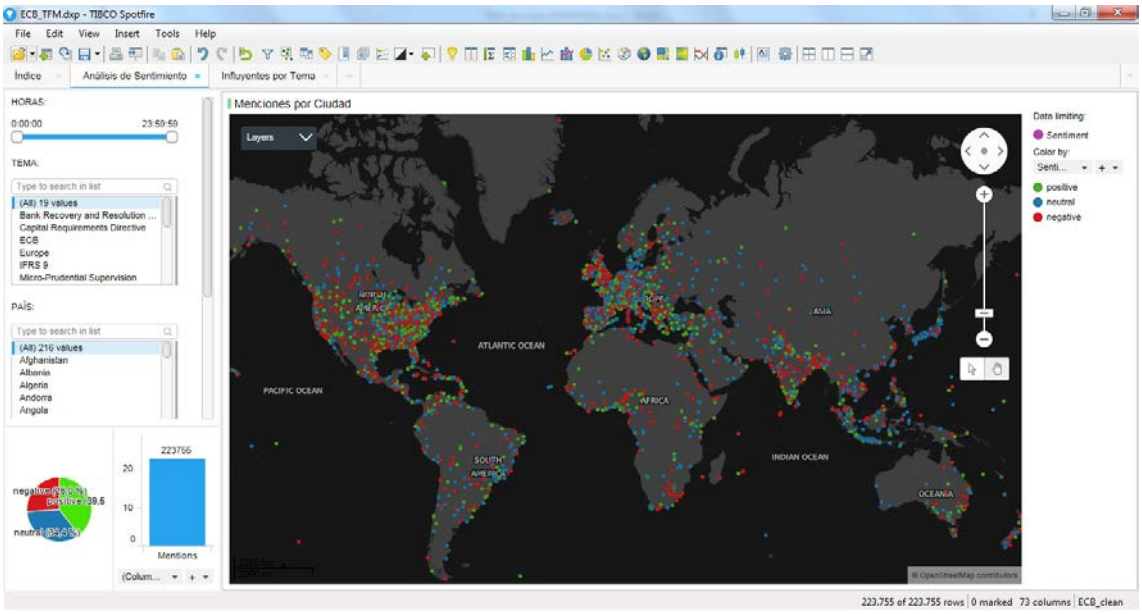
5.6 Visualización y Creación de Dashboards

Spotfire Desktop es una herramienta para crear visualizaciones interactivas y realizar análisis de datos en tiempo real en ordenadores personales o tablets, sin necesidad de un servidor. Con Spotfire Desktop es posible integrar fuentes provenientes de aplicaciones y sistemas empresariales de inmediato, sin apoyo TI.
Spotfire Desktop es una gran manera de aprender acerca de la detección visual de información a medida que explora sus datos. Spotfire Desktop le ofrece una variedad ilimitada de posibilidades para la visualización y la estadística de datos.



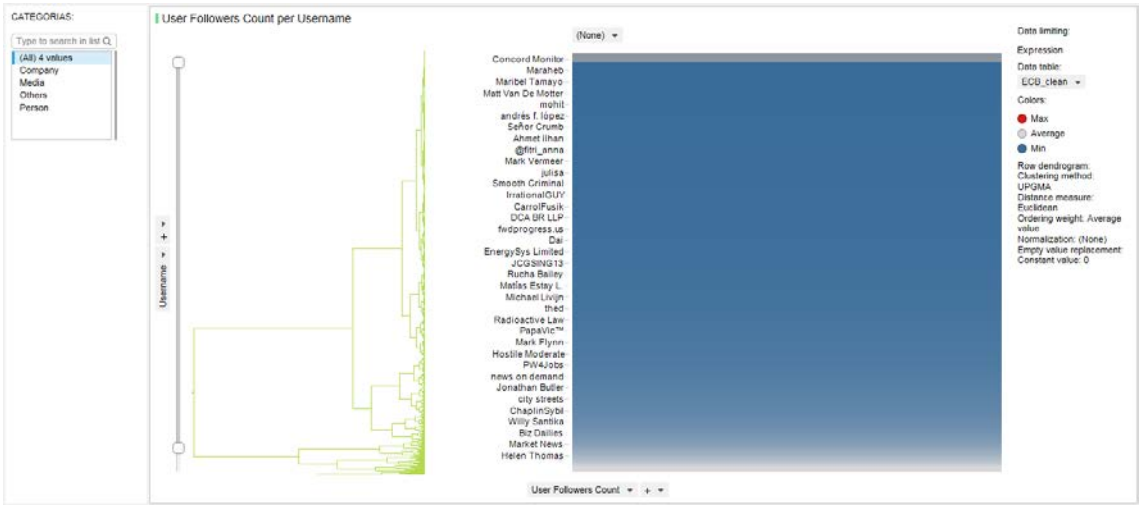
La primera pestaña de la herramienta nos muestra la geolocalización de los tweets clasificados por sentimiento. Geolocaliza las tendencias del momento en cualquier

lugar del mundo. Permite seleccionar un tema e identificar dónde se está hablando de él. Interesante por si se pretende saber qué es lo que le interesa a la comunidad de Twitter de una determinada área geográfica, o identificar en qué lugares del mundo se está hablando de un tema concreto. Muy útil por ejemplo para averiguar dónde se genera y cómo se expande una crisis de social media, o conocer los usuarios más influyentes sobre un determinado tema en un lugar concreto.



La segunda pestaña muestra el top de usuarios por tema, es decir, los usuarios más influyentes por topic. Se ha generado un dashboard en forma de treemap en el cual se puede observar el TOP 10, 20, 50 y 100 de usuarios. Se ha separado por tipo de usuario, no es lo mismo si se trata de una persona, una compañía o un medio.

En la tercera pestaña se muestra el comportamiento de los usuarios dependiendo del número de “followers” que tengan en la cuenta asociada a su nombre. Se ha separado por tipo de cuenta, ya sea persona, media o compañía.



6. CONCLUSIONES

Este trabajo fin de máster ha abordado tareas de análisis automático de los contenidos expresados por los usuarios en las redes sociales, y más concretamente sobre Twitter, una de las más populares en la actualidad.

La plataforma de Social Media;

• El proyecto realiza el rastreo web, utilizando diversas fuentes de información, tales como periódicos, blogs, sitios web, Twitter... uniendo la extracción con la información de las redes sociales.

• En segundo lugar, se investiga a las personas, obteniendo como beneficio el descubrimiento de personas influyentes. Este proyecto se centra en las personas, las empresas y los medios de comunicación.

• El proyecto de Social Media es capaz de agrupar KOL (Key Opinion Leaders), empresas y medios de comunicación.

• Por último, se ha utilizado diferentes motores de análisis para la minería de texto; no se centra sólo en la estadística descriptiva. Por ejemplo, se realizó un análisis de los sentimientos, de KOL, la creación de comunidades entre los temas y personas.

• Se ha desarrollado la capacidad de monitorear el comportamiento social en torno a los topics.

La hoja de ruta seguida en el proyecto ha sido la siguiente:

• Búsqueda por keywords→Búsqueda por fuente de información →Selección de un rango de fechas para realizar las búsquedas →Realización de estadística descriptiva →Realización de análisis estadístico →Identificación de KOL's → Identificación de medios→ Identificación de compañías → Agrupación de KOL´s sobre el comportamiento social.

Funcionalidad:

- Los usuarios pueden descubrir y compartir relaciones de los medios de comunicación, organizaciones, publicaciones. Incluyendo «líderes de opinión» en diferentes canales sociales, se muestran los activos sobre los temas seleccionados (BCE finanzas y cuestiones de reglamentación).
- Se muestran los temas, hashtags y cuestiones en torno BCE y la regulación financiera con el fin de construir toda una ontología para una clasificación efectiva. Se descubren temas, subtemas, hashtags y entidades semánticas relevantes.
- Ideas relacionadas con temas y líderes de opinión clave: descubrimiento y presentación de puntos de vista. Insights visualización.

Documentos entregados:

Documento pdf con explicación sobre el proyecto.
Insights de presentación relacionados con los temas seleccionados y líderes de opinión:
Listas superiores y las métricas básicas.
Influyentes y KOL (personas / organizaciones / medios de comunicación).
Topics principales, métricas de volumen y categorías.
Visualización: Dashboard dinámico.

Eficacia:

El proyecto ha identificado nuevos factores de influencia y temas. Se ha creado una red de personas influyentes por tema.

Mejoras:

Se ha identificado nuevos factores de influencia por tema para mejorar la eficacia del algoritmo. La ampliación de la red de la aplicación del algoritmo de éstos tendría un impacto significativo.

Las perspectivas:

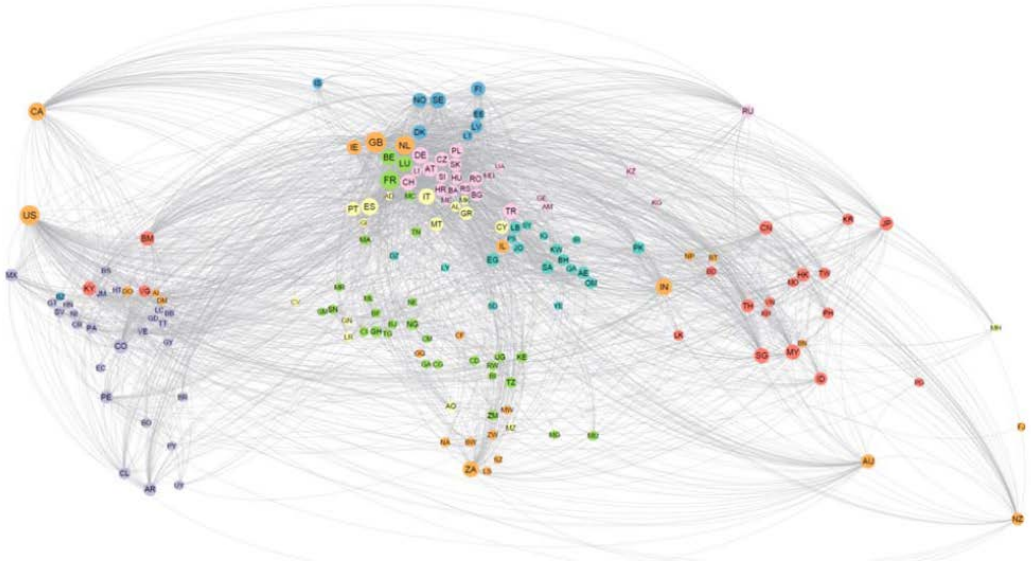
Gran potencial de aplicación de este algoritmo como parte de una metodología para identificar posibles clientes y evolucionar las estrategias de adquisición del banco.

7. VENTAJAS DEL USO DE SNA Y BIG DATA

El descubrimiento de los grupos cohesivos y las comunidades dentro de una red es uno de los temas más estudiados en el análisis de redes sociales (SNA). Ha atraído a muchos investigadores en sociología, biología, informática, ciencia, física, criminología, y así sucesivamente. La detección de una comunidad tiene como objetivo la búsqueda de grupos como subgrafos dentro de una red. Por lo tanto, una comunidad es un

clúster donde muchas aristas enlazan nodos que pertenecen al mismo grupo y otros nodos se convierten en enlaces al crear nodos centrales.
Un enfoque general para la detección de la comunidad consiste en considerar la red como una visión estática en la que todos los nodos y enlaces en la red se mantienen sin cambios durante todo el estudio. Estudios recientes se centran también en la evolución de la comunidad ya que la mayoría de las redes sociales tienden a evolucionar con el tiempo a través de la adición y la supresión de nodos y enlaces. Como consecuencia de ello, los grupos dentro de una red se pueden ampliar o reducir y sus miembros se pueden mover de un grupo a otro en el tiempo.
La mayoría de los estudios sobre la evolución de la comunidad usan propiedades topológicas para identificar las partes actualizadas de la red y caracterizar el tipo de cambios, tales como contracción de la red, cada vez mayor, la división, y fusión. Sin embargo, un trabajo reciente se ha centrado en la evolución / detección de la comunidad por depender enteramente del comportamiento de los miembros del grupo en términos de las actividades que se producen en la red en lugar de tener en cuenta exclusivamente los enlaces y densidad de la red.
Otra característica interesante de las redes sociales es la cohesión de un grupo y la forma en que varía a lo largo del tiempo. De hecho, la cohesión de un grupo es un factor social que evalúa cómo los miembros de un grupo son de cercanos unos de otros, y pueden ayudar a predecir una posible división de la comunidad o desagregación. Esta es una de las razones del porqué es una interesante aplicación nuestro enfoque para explorar los mensajes de Twitter en el análisis de redes sociales.
Para identificar las acciones concretas, la construcción de una red identifica comunidades con sus actores centrales, y muestra el comportamiento de los miembros de la comunidad. Algunos resultados ayudan a comprender el comportamiento de las comunidades en su conjunto o como miembros individuales de tales grupos cohesivos.
Yendo más lejos, podemos utilizar las teorías de probabilidad y posibilidad como dos soluciones alternativas a descubrir las comunidades (perspectiva temporal) y la evolución de la comunidad. Empezando desde instantáneas de la red en diferentes períodos de tiempo, la red social subyacente se analiza con el fin de que primero identifique actores activos (es decir, actores que participan en al menos un número predefinido de actividades) durante una serie de intervalos de tiempo, y luego delimitar las comunidades que forman con el tiempo.
Aparte del hecho de que el enfoque de seguimiento de la evolución de la red identifica comunidades, ofrece una forma básica para identificar a los usuarios, tanto activos como pasivos. El último grupo de usuarios puede ser visto como churners en aplicaciones de gestión de relaciones con los clientes (CRM). Además, el mapeo de las comunidades en una red inicial añade nuevos enlaces que mejoran la accesibilidad a la red, y por lo tanto la circulación del flujo de información.
Mirando en la estructura gráfica que hay detrás de las redes sociales y la clasificación, podemos identificar a los conductores o influyentes en las redes sociales. El núcleo de una red puede verse como una parte central que tiene una alta influencia en los flujos de comunicación que involucran a los otros nodos. De hecho, el comportamiento dinámico o conceptos semánticos de las entidades sociales son una entrada adicional para explotar con el fin de caracterizar y fortalecer de manera significativa una estructura de grupo y poner de relieve su cohesión. La idea clave es que es muy probable que los actores de una red social cambien sus interacciones con el tiempo

mediante la adición o eliminación de relaciones con los demás. Esto tiene un impacto en su posición social en la red y / o su posible afiliación a uno o más grupos sociales. El cambio temporal es en realidad inducido por muchos factores influir cohesión comportamiento actor.



Otro aspecto importante de las redes y gráficos está relacionada con los fenómenos de propagación y como una característica generalizada y significativa de las redes del mundo real. Estos fenómenos son interdisciplinarios influyendo en la ciencia, la ingeniería, las finanzas, los negocios, y en última instancia la sociedad misma. El desarrollo de técnicas de propagación juegan un papel importante en el mantenimiento de las redes existentes y ha permitido, por ejemplo, la sincronización de redes de energía eléctrica, la predicción del comportamiento de sistemas complejos, el descubrimiento de recursos y la supervisión, la localización de las invasiones biológicas y la evaluación de los daños, el control de la propagación del virus y la contención, la descomposición y la inmunización de infraestructura de gran escala social y redes. Mediante el estudio de los procesos de propagación, se puede comprender mejor la información y el conocimiento en los sistemas que a su vez puede dar lugar a algunas mejoras en el rendimiento y la robustez.

Una red social es un grupo de colaboración y / o personas o entidades que se relacionan entre sí y se define formalmente como un conjunto de actores sociales, o nodos, que están conectadas por uno o varios tipos de relaciones.

SNA se ocupa del análisis de redes sociales con el fin de trazar las relaciones, aprender sus significados y aplicar la información inferida entre los miembros de la red. SNA toma prestado muchos conceptos y herramientas de la teoría de grafos porque una red social puede ser vista como un grafo donde los actores están representados por los nodos y las relaciones entre ellos por la bordes del grafo, los coeficientes de ponderación se pueden asignar a los bordes entre los nodos para designar diferentes interacciones fuertes.

La minería de datos de medios sociales es la tarea de los contenidos generados por los usuarios de minería con las relaciones sociales.

Estos datos presentan nuevos retos encontrados en la minería de los medios sociales.

- Big Data Paradox. Los datos de medios de comunicación social son, sin duda grandes. Sin embargo, cuando nos acercamos a individuos para los que, por ejemplo, nos gustaría hacer las recomendaciones pertinentes, a menudo tener pocos datos para cada individuo en particular. Tenemos que aprovechar las características de los medios de comunicación social y utilizar su multidimensional, es decir, utilizar los datos de múltiples sitios con estadísticas suficientes para la explotación minera eficaz.
- La obtención de muestras suficientes. Uno de los métodos utilizados comúnmente para obtener muestras suficientes es recoger los datos es a través de interfaces de programación de aplicaciones (API) de medios sociales. Sólo se pueden obtener una cantidad limitada de datos todos los días. Sin saber la distribución de la población
- Eliminación de ruido. En la literatura clásica de minería de datos, una exitosa eliminación de ruido en los datos parece una falacia. El ejercicio de la minería implica extensos pre-procesamiento de datos y eliminación de ruido como "garbage in and garbage out", es decir, “Basura dentro, Basura fuera”.
  Por su naturaleza, los datos de medios sociales pueden contener una gran parte de datos ruidosos. Para estos datos, nos damos cuenta de dos observaciones importantes:
  - La eliminación de ruido a ciegas puede empeorar el problema planteado en la gran paradoja de datos porque la eliminación también puede eliminar información valiosa.
  - La definición de ruido se hace complicada y relativa porque depende de nuestra tarea en cuestión.
- Evaluación dilema. Es un procedimiento estándar de evaluación de los patrones de evaluación en la minería de datos, consiste en tener algún tipo de realidad del terreno. Por ejemplo, un conjunto de datos se puede dividir en entrenamiento y conjuntos de prueba. Sólo los datos de entrenamiento se utiliza en el aprendizaje, y los datos de prueba sirven para testear las pruebas. Sin embargo, la parte test a menudo no está disponible en la minería de los medios sociales. Evaluando los patrones de la minería en los medios de comunicación social se plantea un desafío aparentemente insuperable. Por otro lado, sin una evaluación creíble, ¿cómo podemos garantizar la validez de los patrones?

El análisis de los gráfos a menudo implica el cálculo de un número de métricas. Un grupo de métricas son las medidas de centralidad. Estas son las cantidades que ilustran la importancia de cada vértice dentro de la red. La importancia puede ser con respecto al parentesco u otros actores dentro de la red, siendo mínimamente separado de todos los demás agentes. Los más comunes son la centralidad de grado, centralidad, vector propio centralidad y la centralidad de intermediación. Se observa que los indicadores de centralidad en general no están de acuerdo en que vértice tiene la puntuación más alta, por lo tanto, que se considere más importante.

8. BENEFICIOS DEL USO DE SNA

El análisis de redes sociales es utilizado en una amplia gama de aplicaciones y disciplinas. Algunas aplicaciones de análisis de red comunes incluyen la agregación de datos y la minería, establecer modelos de propagación de la red, modelado de la red y toma de muestras, atributo de usuario y análisis de comportamiento, el apoyo de recursos mantenidos en la comunidad, el análisis de la interacción basada en la localización, el intercambio social y filtrado, el desarrollo de los sistemas de recomendación, y la predicción de enlace y resolución de entidades. En el sector privado, las empresas utilizan el análisis de redes sociales para apoyar actividades como la interacción con el cliente y el análisis, información de marketing, análisis del desarrollo del sistema, y las necesidades de inteligencia de negocios. Algunos sectores públicos lo utilizan para incluir el desarrollo de estrategias de participación líder, el análisis de la participación individual y de grupo y uso de los medios, y la resolución de problemas basados en la comunidad. El análisis de redes sociales también se utiliza en las actividades de inteligencia y contra-inteligencia. Esta técnica permite a los analistas para asignar de forma clandestina o encubierta una red de espionaje, una familia del crimen organizado o de una banda callejera. La Agencia Nacional de Seguridad (NSA) utiliza sus programas de vigilancia electrónica de masas de forma clandestina para generar los datos necesarios con el fin de llevar a cabo este tipo de análisis sobre las células terroristas y otras redes que se consideren relevantes para la seguridad nacional. La NSA utiliza hasta tres nodos de profundidad durante este análisis de redes. Después de la asignación inicial de la red social, se lleva a cabo un análisis para determinar la estructura de la red y determinar, por ejemplo, los líderes de la red. Esto permite que los activos militares o policiales lancen ataques de captura en un objetivo de alto posicionado en liderazgo para interrumpir el funcionamiento de la red. La NSA ha estado llevando a cabo análisis de redes sociales en Call Detail Records (CDR), también conocidos como metadatos desde poco después de los ataques del 11 de septiembre. Grandes corpus textuales se puede convertir en redes y después se analizan con el método de análisis de redes sociales. En estas redes, los nodos son los actores sociales, y los enlaces son acciones. La extracción de estas redes se puede automatizar, mediante el uso de programas de análisis.

Las redes resultantes, pueden contener miles de nodos, entonces se analizan mediante el uso de herramientas de la teoría de red para identificar a los actores clave, las comunidades o las partes clave, y general se estudian propiedades tales como la robustez o la estabilidad estructural de la red global, o centralidad de ciertos nodos. Esto automatiza el enfoque introducido por el Análisis Cuantitativo de Narrativa, con lo cual trillizos sujeto-verbo-objeto se identifican con los pares de actores vinculados por una acción, o pares formados por el actor y objeto.

<u>Beneficios en el área de Finanzas:</u>

1. ANALIZAR EL SENTIMIENTO PÚBLICO HACIA LA ECONOMÍA:
Un análisis de opinión pública en general puede decir a los bancos desde el principio si la gente es optimista sobre el desarrollo económico y quieren gastar dinero o si se sienten inseguros y por lo tanto son más propensos a ahorrar sus ingresos. Siguiendo

estas conversaciones, los bancos pueden adaptar y perfeccionar su estrategia de negocio de acuerdo con estas tendencias.
La inteligencia de datos sociales ofrece información en tiempo real que se pueden entregar en el momento adecuado a los servicios de un banco. Mediante el control de diferentes frases y términos clave, por ejemplo, el desempleo se puede relacionar con menciones o discusiones acerca de préstamos para la vivienda y las hipotecas. El uso de estas ideas da a los bancos un nuevo enfoque que los especialistas pueden utilizar, aportando más información a las estadísticas "oficiales" y los registros. La recogida de datos para los registros a menudo toma mucho tiempo, por lo que los datos quedan casi obsoletos en el momento en que se ponen en las manos de los bancos e instituciones financieras. Por el contrario, los datos sociales se pueden recoger de forma rápida y reflejan los desarrollos actuales en tiempo real.

2. ENTENDER OPINIÓN DE LOS CONSUMIDORES SOBRE SUS PRODUCTOS:
El control de la conversación acerca de sus productos, así como sus competidores permite a los bancos ver lo que más importa a sus clientes y cómo sus productos se perciben en comparación con los demás. La inteligencia social permite a los bancos tomar decisiones bien fundamentadas sobre los aspectos de sus productos, si necesitan mejoras, y que productos completamente nuevos valen la pena desarrollar en el mercado.

3. IDENTIFICAR OPORTUNIDADES DE INTERACCIÓN CON EL CLIENTE:
El control de la conversación en torno a estos hashtags ayuda al banco para segmentar su base de clientes potenciales en diferentes grupos con diferentes necesidades y objetivos. Diferenciar claramente entre esos grupos permite a los bancos orientar sus estrategias de comunicación o de marketing específicamente a aquellos segmentos de los clientes y ofrecerles exactamente los productos adecuados en diferentes etapas de su vida.

4. PLANIFICACIÓN Y PUESTA A PUNTO DEL LANZAMIENTO DE PRODUCTOS:
Los bancos y otras instituciones financieras lanzan nuevos productos sobre una base regular. Con el fin de averiguar qué características deben tener estos productos para servir mejor a las necesidades de los clientes, los bancos pueden depender de monitoreo de medios sociales para darles una imagen más clara.

Al final, hay una enorme oportunidad de utilizar SNA en algunos de nuestros proyectos diarios:

- Los usuarios están tratando cada vez mayores conjuntos de datos y se necesitan a nivel hora, las capacidades que pueden ayudar a filtrar la información de red más rápido y con más eficiencia.
- Los usuarios necesitan identificar rápidamente a individuos / grupos cruciales para una mejor optimización de los recursos limitados debido al dinamismo de las redes de destino.
- Identificar las características de las redes (que no son atractivas) y analizar cómo estas redes son dinámicas en el tiempo.

- Los usuarios saben que todas las relaciones o conexiones en una red social no son iguales y métodos tales como la ponderación en las relaciones de las personas deben ser utilizados para estudiar el impacto de tales las relaciones de la red.

El análisis de Redes Sociales está siendo utilizado en varios campos. Estudiar las citas y afiliaciones de las personas puede ser valioso para el descubrimiento de varios patrones y anti-patrones, incluidos el uso fraudulento de tarjetas de crédito o robo, las reclamaciones de seguros falsas, el abuso de la asistencia sanitaria, uso de información privilegiada, etc. De acuerdo con Dan McKenzie, especialista en soluciones de fraude de SAS Canadá, las instituciones financieras están descubriendo que SNA está ayudando a destapar las actividades de 20 a 50 veces más fraudulentas que antes de usar SNA.
Las actividades fraudulentas son difíciles de capturar, ya que los incidentes son a menudo enterrados bajo enormes cantidades de las actividades normales. Por otra parte, las actividades están a menudo disfrazadas dentro de actividades normales.
Las aplicaciones generales de SNA son:

- Para una mejor orientación al cliente, se realizan promociones potenciales basándose en su historial de compras.
- En la identificación de clientes fieles que son vocales, activos y apasionados. Se pueden caracterizar como embajadores de la marca.
- En la reducción de las tasas promedio de deserción en la industria de las telecomunicaciones mediante la identificación de los conectores centrales y ofrecer premios especiales o experiencias personalizadas.
- En la lucha contra las actividades terroristas mediante la caracterización de las organizaciones de la red para determinar la probabilidad y el impacto de la actividad terrorista.
- En la detección de fraude de atención médica mediante la detección de patrones, el establecimiento de vínculos entre individuos, y para conectar las relaciones no evidentes.

En nuestras actividades diarias, SNA podría ser utilizado desde un punto de vista comercial, para:

1. Análisis de Redes Sociales para la creación y uso de la inteligencia de clientes:
   Las organizaciones quieren utilizar los datos de los medios sociales para entender el comportamiento y las necesidades de sus clientes o grupos destinatarios específicos de personas con respecto a los servicios o productos actuales o futuros de la organización. Sugiriendo tres enfoques principales para la observación de los medios sociales - herramientas para la presentación de informes a través de diversos canales, sistemas de puntuación de las tarjetas de información general, y las técnicas de análisis predictivo. Se discute también sobre el cuarto enfoque, es decir, utilizar una plataforma analítica predictiva que combina la minería de textos y análisis de redes con otros métodos de predicción y agrupación.
   La combinación de la red y la minería de texto revelan nuevos puntos de vista variados en el comportamiento del cliente en los medios de comunicación social,

que no habría sido posible de otro modo mediante el uso de cualquiera de estas técnicas por sí solas. La combinación de estructuras de referencia con el análisis de los sentimientos de los mensajes de diversos foros en línea de la red permitió el estudio del posicionamiento de los usuarios positivos y negativos en la relación con su peso relativo como seguidores y personas influyentes en el foro de discusión subyacente. Una buena comprensión de los segmentos de los medios de comunicación social puede hacer una valiosa contribución a la decisión sobre cómo dar forma e invertir en planes de comercialización y medios sociales de la organización.

2. Análisis de Redes Sociales en el cambio organizacional:

El análisis de Redes Sociales ayuda a revelar las redes estratégicamente importantes que no se pueden encontrar en los organigramas formales. Es compatible con el descubrimiento de las estructuras subyacentes informales que existen en las organizaciones.

SNA utilizada dentro de una organización también puede ser denominada como el Análisis de Redes de organización. En este enfoque, el punto central es la identificación de las redes cruciales, dentro de los límites de la organización, la comprensión de la estructura de las relaciones personales y de grupo dentro de las redes, y el uso de ese conocimiento para mejorar el rendimiento del negocio. Los buenos gerentes comprenden el papel desempeñado por estas redes y cómo utilizarlos al máximo.

Retomando a SNA para gestionar el cambio organizacional se ha visto obstaculizado por la idea errónea de que es altamente conceptual y el conocimiento obtenido es difícil de transformar en acciones pragmáticas. Esto se debe a estudios de caso académicos que muestran las oportunidades en lugar de los resultados de negocio que se pueden obtener a partir de SNA. Con el fin de reforzar el punto de vista comercial, los casos de uso del negocio se agrupan en tres grandes categorías de la siguiente manera:

- La gestión de los recursos humanos en las grandes empresas por SNA para trazar y medir las relaciones entre las personas que no eran visibles para estudiar cómo la responsabilidad, la influencia y el poder se difunden a través de grandes grupos de personas.
- Gestión de Procesos de Negocio a través de SNA que puede dar una idea de negocio en el funcionamiento de sus mejores empleados.
- La reestructuración estratégica mediante el uso de métricas y herramientas apropiadas SNA en varios niveles, como el individuo, la organización y la industria. Son útiles para los programas de reestructuración y cambios organizativos.

3. Análisis de Redes Sociales para el comportamiento de la salud, el entendimiento de SNA proporciona una perspectiva de cómo funciona una sociedad. Las redes sociales tienen una enorme influencia en el comportamiento de la salud de los individuos. Los resultados del análisis de redes sociales pueden ser utilizados por el Gobierno para el diseño de planes de salud, beneficios y que tomen medidas preventivas durante algunos brotes

de enfermedades. Las compañías farmacéuticas pueden dirigirse a grupos demográficos y mercados específicos. Las compañías de seguros pueden diseñar sus planes de seguro de una mejor manera.

9.BIBLIOGRAFÍA

Repositorios de datos sociales

- Standford Large Network Dataset Collection http://snap.stanford.edu/data/index.html
- Social Computing Data Repository at ASU http://socialcomputing.asu.edu/pages/datasets
- KDNuggets Datasets for Data Mining and Data Science http://www.kdnuggets.com/datasets/index.html
- Trust network datasets http://www.trustlet.org/wiki/Repositories_of_datasets
- Data-Science central http://www.datasciencecentral.com/profiles/blogs/20-free-big-data-sources-everyone-should-check-out
- QUANDL (Datos económicos y financieros + social media) https://www.quandl.com

Referencias

Sobre Minería de datos de Redes Sociales - Mining the Social Web (libro) - Social Network Analysis for Startups

Sobre Análisis de Sentimiento - Bing Liu. Sentiment Analysis and Opinion Mining, Morgan & Claypool Publishers, May 2012.

Sobre Influencia - Fluent: The Razorfish Social Influence Marketing Report.
The Klout Score Methodology Secrets Revealed

Sobre comunidades en redes sociales - How we analyzed Twitter social media networks with NodeXL

A. Sobre bases de datos de grafos:
   - Wikipedia: http://en.wikipedia.org/wiki/Graph_database
   - Libro: Graph Databases (O'Reilly) http://graphdatabases.com
   - Ranking de bases de datos de grafos: http://db-engines.com/en/ranking/graph+dbms
B. Sobre librerías de análisis:
   - Igraph: Statistical Analysis of Network Data with R (libro) http://www.amazon.com/Statistical-Analysis-Network-Data-Use/dp/1493909827/
   - GraphX: A gentle introduction to GraphX in Spark http://www.sparktutorials.net/analyzing-flight-data:-a-gentle-introduction-to-graphx-in-spark
C. Sobre visualización:
   - Gephi:
     - Learn how to use Gephi https://gephi.org/users/

- **Introduction to Network Analysis and Visualization**http://www.martingrandjean.ch/gephi-introduction/

PYTHON:

http://doc.scrapy.org/en/1.0/topics/spiders.html#crawlspider

http://doc.scrapy.org/en/1.0/topics/items.html#module-scrapy.item

http://doc.scrapy.org/en/1.0/topics/feed-exports.html#feed-exports

https://www.crummy.com/software/BeautifulSoup/bs4/doc/#calling-a-tag-is-like-calling-find-all

http://jarroba.com/scraping-python-beautifulsoup-ejemplos/

Algunas referencias sobre R:

·Manuales en CRAN -An introduction to R
-R Data Import/Export
-R Reference Card
-Using R for Data Analysis and Graphics

·Páginas Web -Quick-R página web muy sencilla con explicaciones rápidas
-R by Example colección de scripts de R para aprenderlo con ejemplos

·Libros -The R Book, Michael J. Crawley (Wiley)
-R in a Nutshell, J. Adler (O'Reilly)
-R in Action, Robert I. Kabacoff (Manning)
-Machine Learning for Hackers, Drew Conway, John Myles White (O'Reilly)
-Applied Predictive Modeling, Max Kuhn, Kjell Johnson (Springer)

Algunas referencias sobre igraph:

·Material online -The igraph book (incompleto)
-igraph wikidot
-Manual sencillo en español
·Libros -Statistical Analysis of Networ Data with R